Homecomputer

1 DEUTSCHLANDS ERSTE HEIMCOMPUTER-ZEITSCHRIFT

3. Jahrgang

Januar '85, 5,50 DM, 48 öS, 6,00 sfr



Aktuell: Homecomputer unter neuer Regie!

Seite 2

Commodore 64
VC-20
ZX-81
ZX-Spectrum
TI-99/4 A
Apple II
Atari
Laser 2001

Neu!
Software-Service

Tips & Tricks

- Data-Generator
- Maschinenprogramm ohne Monitor
- Video Controller

u. v. m.

Report:

Disketten
Einmaleins für Floppy-User
Handheld-Computer-Kombination

Top-Programm Reaktor VC-64

AKTUELL

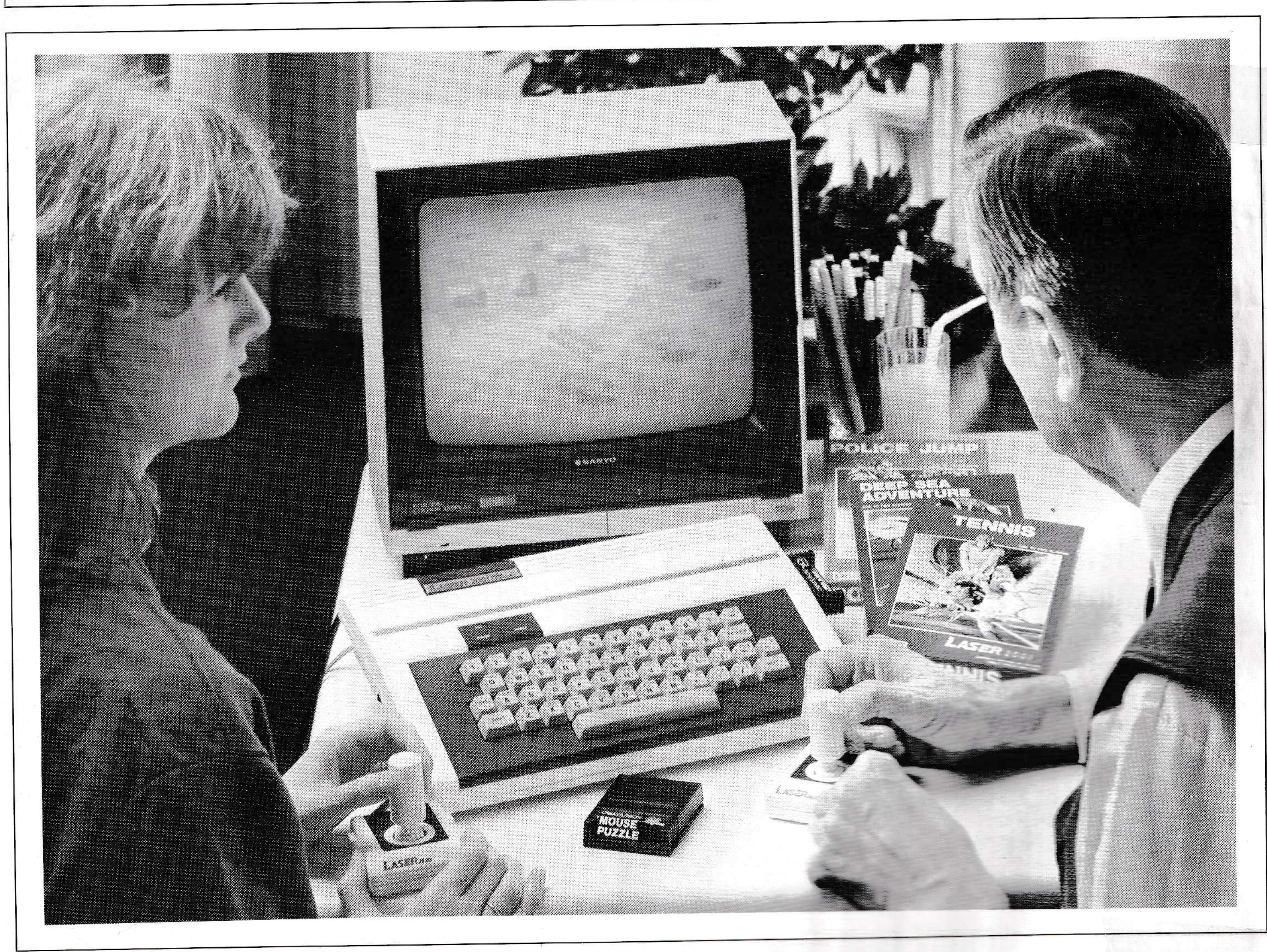
Diese Ausgabe des „Homecomputer“ hat das Team vom Tronic-Verlag gestaltet. In der nächsten Ausgabe lesen Sie, liebe „Homecomputer-Freunde“, was sich ändern wird, was wir besser machen wollen, was wir alles drauf haben.

Wir haben zwischenzeitlich viele Anregungen, aber auch Kritik gehört. Wir bitten Sie, zu bedenken, daß der Tronic-Verlag lediglich die Titelrechte, nicht aber den bisherigen Verlag der Zeitschrift „Homecomputer“ erworben hat.

Wir werden „Homecomputer“ so attraktiv wie möglich gestalten, damit Sie uns, liebe Leser, treu bleiben.

Im nächsten Heft werden wir das neue Konzept vorstellen.

Redaktion Tronic-Verlag
3444 Wehretal 1



INHALT

Homecomputer

Heft Januar 1985 – Jahrgang 3

Report

Uhren, Datenbank, Handheld Report 6

Disketten:
Kleines Einmaleins für Floppy-User 9

Tips & Tricks

Maschinenprogramm ohne Monitor VC-64
Data-Generator VC-64
Video Controller VC-20 5

Reset ohne Programmverlust VC-64
Runden von Zahlen VC-64
Paddle am VC-64 66

Homecomputer-Bibliothek

Lesestoff für jeden 30

Review

Atari:
Das Geheimnis von Atlantis 79

Software

Top-Programm des Monats

VC-64
Reaktor (Simons Basic) 12
Concentration 20
Datenbank 50

Apple II
Maskengenerator 23
Labyrinth 26
Data-Generator 29

TI-99
Lift-Bär 32
Luftkampf 36

Laser 2001
Clipper-Flipper 39

Atari
5 gewinnt 46
Maniac 67

VC-20
Geldsammler 56
Zyklo 58

ZX-Spectrum
Plong 61

ZX-81
Irrgarten 3D 54
Grand Prix 63

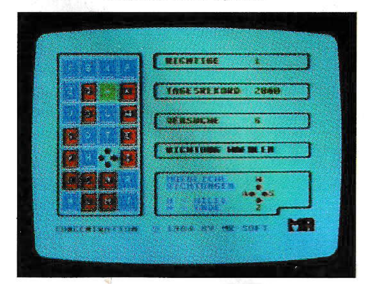
Kleinanzeigen 74

Kassetten-Service

Neu in Homecomputer 76

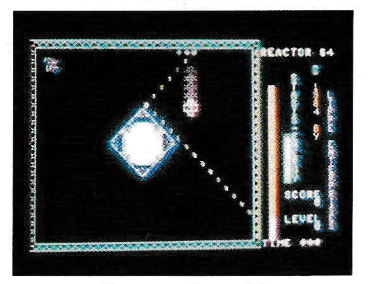
Top-Programm des Monats

Concentration



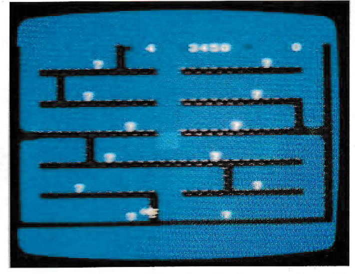
Commodore 64

Reaktor



Commodore 64

Lift-Bär



TI-99

Das Softwaremagazin für Freizeit und Hobby



Jetzt überall im Zeitschriftenhandel

STOP + + + NEUE AUSGABE + + + STOP

Tips + Tricks

Commodore 64

Maschinenprogramm ohne Monitor abspeichern
Möchte man ein Maschinenprogramm, Grafik oder Sprite ab einer bestimmten Adresse abspeichern, so gibt es im Basic-Befehlssatz keine Möglichkeiten dazu. Es ist zwar mit einem kleinen Basic-Programm möglich, jedoch sehr umständlich und sehr zeitaufwendig. Mit einem kleinen Trick ist es jedoch ganz einfach. Man setzt den Basic-Start (in 43/44) auf die Startadresse und den Variablen-Start auf die Endadresse (in 45/46).

Die folgenden Zeilen ermöglichen dies:
AD = Anfangsadresse
POKE 43, AD AND 256:
POKE 45, ED AND 256:

ED = Endadresse
POKE 44, INT (AD/256)
POKE 46, INT (ED/256)

Nach diesen „POKE'S" kann das Maschinenprogramm mit einem normalen SAVE auf Disk oder Cassette gespeichert werden. Mit LOAD „name", 8,1 bzw. LOAD „name", 1,1 wird es wieder geladen.

STOP-Taste ausschalten
Will man Programme schützen, so gibt es die Möglichkeit mit POKE 788,52 (zu Beginn des Programms) den IRQ-Vektor zu verändern. Das Programm läßt sich dann nicht mehr mit STOP unterbrechen. Die eingebaute TI$-Uhr bleibt in diesem Fall stehen. Durch POKE 788,49 wird beides wieder aktiviert.

Zeichen an beliebige Stelle setzen
Leider kennt der C 64 keine Zeichen-Positionierbefehle. Die folgenden Zeichen helfen diesen Mangel zu beheben.
HT = Horizontale Position 0 bis 39
VT = Vertikale Position 0 bis 24
10 BZ = 1024 + 40 * VT: POKE 209, BZ AND 255: POKE 210, BZ/256: POKE 211, HT 20 PRINT „ZEICHEN"
Die Zeilen kann man sehr gut als Unterprogramm verwenden und somit an jede beliebige Stelle des Bildschirmes den Cursor setzen.

VC-20

Video Controller
Im Gegensatz zu vielen anderen Computern besitzt der VC-20 einen Video-Controller. Dieser Controller übernimmt die gesamte Aufbereitung des Fernsehbildes. Da dieser Controller ab Adresse 36864 ($ 9000) erreichbar ist, lassen sich kaum vorstellbare Effekte erzielen. Will man beispielsweise den Bildschirm verschieben, so ist dies für den Controller keine Schwierigkeit. Mit dem Befehl POKE 36864,X teilt man ihm die horizontale Position und mit POKE 36865,X die vertikale Position mit. Mit diesem Trick ist es möglich, das Bild an jede Position des Schirmes zu verschieben.

DATA-Generator für VC-64

```
10 PRINT"□□ *** DATA GENERATOR     F.BRALL ***  □□":POKE53281
,8
11 INPUT"□ZEILENNR. (>22) ";ZN:INPUT"□ANFANGSADRESSE ";A
12 INPUT"ENDADRESSE        ";E
13 PRINT"□10 FORI="A"TO"E":READ DA:POKEI,DA:NEXTI":PRINT"ZN=
"ZN":E="E":A="A;
14 PRINT":GOTO 15":GOTO20
15 PRINT"□";:IFA>ETHEN21
16 PRINT"□"ZN"DATA ";:FORO=1 TO 17:D=PEEK(A):D$=STR$(D):S$="
"
17 FORI=1 TO 4:E$=MID$(D$,I,1):IF E$<>" "THEN S$=S$+E$
18 NEXTI:PRINTS$",";:A=A+1:IFA>E THEN O=17
19 NEXTO:PRINT"▮▮ ":PRINT"■A="A":ZN="ZN+1":E="E":GOTO 15"
20 POKE 631,19:FORI=632TO640:POKEI,13:NEXT:POKE198,9:END
21 PRINT"□";:FORI=11 TO 17:PRINTI:NEXT:PRINT"GOTO 22":GOTO 2
0
22 PRINT"□";:FORI=18 TO 22:PRINTI:NEXTI::PRINT"LIST":GOTO20
```

In vielen Basic-Programmen werden Maschinenroutinen verwendet, um die nötige Geschwindigkeit zu erreichen. Diese Routinen werden dann oft als „DATA-Zeilen" in das Programm eingefügt.
Wer jedoch schon einmal ein längeres Maschinenprogramm programmiert hat, weiß, wie mühselig es ist, das Programm in „DATA-Zeilen" zu übersetzen.

Das hier vorgestellte Programm nimmt uns die gesamte Arbeit ab. Hat man ein Maschinenprogramm im Speicher stehen, so lädt und startet man den DATA-Generator. Nach der Eingabe der ersten Zeilennummer sowie Anfangs- und Endadresse des Maschinenprogramms erstellt das Programm selbständig die erforderlichen „DATA-Zeilen" sowie die erforderliche Einlese-Schleife. Bei längeren Maschinenprogrammen kann dies einige Minuten dauern. Nach der Erstellung löscht sich der DATA-Generator selbst, so daß nur die DATA-Zeilen im Speicher bleiben. Man kann nun das Maschinenprogramm wie ein Basic-Programm laden und speichern. Selbstverständlich können auch Sprites oder Bildschirminhalte u. v. m. in „DATAS" umgesetzt werden. Obwohl der DATA-Generator nur sehr wenige Bytes belegt, wird er sich schnell zu einem unentbehrlichen Werkzeug verwandeln.

Uhren-/Datenbank-/Handheld-Computer-Kombination

Computer-Innovation und Miniaturisierung sind nicht nur seit geraumer Zeit in aller Munde, sondern werden vielfach als Synonyme behandelt: man hat sich an den Gedanken gewöhnt, daß Datensysteme nicht nur leistungsfähiger, sondern auch platzsparender werden.
Und das sicher auch mit gutem Grund! Was nutzt mir der beste Rechner, wenn er stets zuhause bleiben muß?. Computer benötigt man schließlich nicht nur im Wohnzimmer.
Derlei grundsätzliche Überlegungen machen einen wesentlichen Reiz aller Neuerscheinungen im Handheldbereich aus. Homecomputer freute sich daher auch über eine Presseinformation des renommierten japanischen Uhren-Herstellers SEIKO, der auf ein neues Produkt aufmerksam macht, das "wrist module" UC-2000.
Stefan Kaus hat sich die sensationelle "Computeruhr" näher angeschaut – hier sein Erfahrungsbericht:

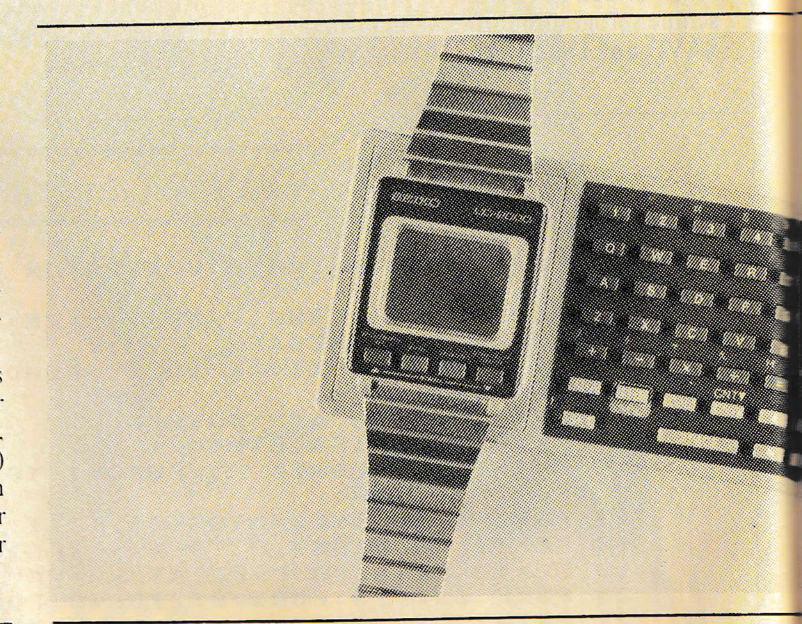
Seit sich die LCD-Armbanduhren allgemein durchgesetzt haben, warten die einschlägigen Anbieter immer wieder mit Zusatzfunktionen auf, die zum Teil nützlich, zum Teil aber auch überflüssige Spielerei darstellen. Was für einen Gewinn bringt mir zum Beispiel eine Armbanduhr, die über einen eingebauten Kalender verfügt, der alle Schaltjahre bis 2100 berücksichtigt, wenn wir 1985 schreiben und die Lebenserwartung eines Durchschnittseuropäers bei 75 Jahren liegt? Worin liegt der Wert eines Chronometers, der 99 Ortszeiten verfügbar macht, wenn ich einmal im Jahr nach Mallorca fliege und ansonsten nur zwischen Sommer- und Winterzeit hin und her zuschalten habe?
Deshalb war ich auch recht skeptisch, als ich meine neue Seiko-Uhr in Händen hielt. Ach, wieder so eine, mit der man drei Rufnummern speichern kann, dachte ich. Um so überraschter war ich dann von der Leistungsfähigkeit des gesamten Systems.

UC-2000

Die Kürzel "UC-2000" "UC-2100" und "UC-2200" stehen für eine Armbanduhr mit 4-zeiligem LCD-Display, eine Westentaschen-Tastatur sowie einen kleinen Tischrechner mit eingebautem 20-Zeichen-Thermodrucker.
Die Uhr dient einerseits als Handgelenkcomputer, der **ständig** T a g, M o n a t, W o c h e n t a g (Klartext) und U h r z e i t anzeigt, zum anderen als Display für Mini-Tastatur und Rechner sowie als Datenbank.

Näheres zum Aufbau

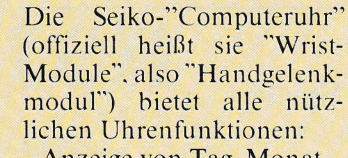
Kernstück des Systems ist Die Armbanduhr "UC-2000", die die Größe eines normalen Chronometers aufweist. Auffälliges Merkmal ist ihr Display, bestehend aus 4 Zeilen a 10 Zeichen, das alle dargestellten Buchstaben, Ziffern und Sonderzeichen aus einer



5x7 Punkt-Matrix aufbaut.
Unterhalb des Diplays befindet sich eine Vierergruppe von Funktionstasten, die deutlich beschriftet und ohne Hilfsmittel (Kugelschreiber, aufgebogene Büroklammer o.ä.) bedient werden können.
Das unempfindliche Resin-Gehäuse macht die Uhr extrem leicht und angenehm tragbar. Die Display und Tasten umgebende Frontplatine ist blendfrei und bildet einen modischen, wenn auch nicht poppigen Kontrast zum elfenbeinfarbenen Gehäuse-Mittelteil. Die Rückseite besteht aus einem verschraubten Metallplättchen, das ein Öffnen der Uhr und somit auch einen Batteriewechsel durch den Besitzer erlaubt. Aus eigener Erfahrung weiß ich aber,daß

bei derartigen Eingriffen Vorsicht geboten ist: die Batteriehalterung ist sehr empfindlich, so daß man sich ohne geeignetes Werkzeug lieber auf den Fachhändler verlassen sollte.
Das Edelstahlarmband ist stufenlos verstellbar, wodurch die Uhr jedem Armumfang angepaßt werden kann.
Als Prozessor besitzt die UC-2000 eine 4bit-CPU, die dementsprechend langsam ist, für die Belange des Benutzers jedoch genügt. Nachteilig macht sie sich lediglich in Zusammenwirken mit dem Tischgerät "UC-2000" bemerkbar (vgl. unten).
Die Seiko UC-2000 verfügt über einen 2K-RAM (2 Blöcke a 1K) sowie einen 6K-ROM für das Betriebssystem und 1.5K-ROM Zeichengenerator.
Ein piezoelektrischer Minilautsprecher gibt Alarm, Stundenschlag und Rückmeldung bei Tastatur-Eingaben. Angetrieben wird das ganze durch eine Lithium-Zelle, für die Seiko eine durchschnittliche Lebensdauer von 18 Monaten angibt.

Funktionen



Die Seiko-"Computeruhr" (offiziell heißt sie "Wrist-Module", also "Handgelenkmodul") bietet alle nützlichen Uhrenfunktionen:

- Anzeige von Tag, Monat, Wochentag, Stunde, Sekunden s t ä n d i g!
- Alarmfunktion (abschaltbar)
- Chiming (Stundenschlag)
- Stoppuhr mit 4-zeiliger Anzeige: betriebsbereit meldet sich die Uhr wie folgt

1. Zeile: STOPWATCH
2. Zeile: 0:00'00" 00
3. Zeile
4. Zeile: 0:00'00" 00

Wenn Sie nun starten, wird Zeile 1 gelöscht, in Zeile 3 wird "RUN" gemeldet, und in Zeile 4 wird die aktuelle Zeitnahme eingeblendet. Bei Zwischenzeitnahme meldet Zeile 1 "SPLIT" und Zeile 2 zeigt die exakte Messung an: die Zeilen 3 und 4 bleiben davon unberührt, zeigen ihre Informationen weiterhin an und bieten somit ein Höchstmaß an Komfort.

Die fünfte Funktion, nämlich die Bereitstellung einer persönlichen Datenbank, macht allerdings die Mini-Tastatur UC-2100 erforderlich. Dieses kleine Extra verhilft dem UC-2000 zu einer vollwertigen alphanumerischen Tastatur (Qwerty) mit 61 Tasten sowie einer Reihe nützlicher Shift-Funktionen.
Die UC-2100 ermöglicht Ihnen den Zugriff auf die beiden RAM's, die mit "MEMO A" und "MEMO B" etikettiert sind und 2000 Zeichen nicht näher spezifizierten Speicherplatz parat stellen.
Der eigentliche Clou ist allerdings der Anschluß Ihrer Armbanduhr an die Eingabeeinheit. Da gibt es weder Schnittstelle noch Kabel, keine Stecker oder Klinkenbuchse, sondern lediglich ein quadratisches Feld links neben den Tasten, auf dem Sie Ihre UC-2000 festrasten können – und schon die Verbindung zur Tastatur hergestellt haben. Eine Induktionsschleife sorgt für einwandfreien Datentransfer.
Sie wählen nun Block A und schreiben Ihre Notizen in den jeweiligen RAM. Damit derartige Eintragungen so ökonomisch wie möglich gestattet werden können, stellt Ihnen der ROM-residente Zeichengenerator eine ganze Reihe nützlicher Piktogramme zur Verfügung: dazu zählen ein Telefonhörer, ein Flugzeug, ein Weinglas, ein eiliger Zeitgenosse und die Darstellung eines kräftigen Handshaking ebenso wie die Spielkartenfarben und diverse Sonderzeichen. Notierte Termine sind auf diese Weise leicht zu spezifizieren und die RAMs optimal auszunutzen.
Die Eingabe-Einheit UC-2100 dient des weiteren auch zur Anwahl des CAL-Modus, wodurch die Standard-Rechenfunktionen (Grundrechnungsarten) auf dem Uhren-Display verfügbar sind.
Dabei werden die jeweiligen Operationen recht komfortabel angezeigt, wie unser Beispiel zeigt:

Aufgabe:

15,67+3,98+44,47+2,25-66,30x4

Anzeige:

15.67	19.65	64.12	66.37
+ 3.98	+ 44.47	+ 2.25	- 66.30
0.07			
x . 4			
= 0.28			

Da die UC-2100 Tastatur ebenso breit ist wie ein handelsüblicher Kugelschreiber lang und die gleiche Tiefe wie ein Lineal besitzt, ist sie leicht in jeder Westentasche unterzubringen. Man sollte sie in jedem Fall dabei haben, damit die Speicherfunktionen der Armbanduhr UC-2000 voll genutzt werden können.
Ich will allerdings erwähnen, daß man gespeicherte Inhalte der Blöcke A und B auch ohne Tastatur seiner Uhr entlocken kann: Man drückt einfach die Taste "MODE" unterhalb des Displays und kann Block A mittels "SCROLL UP" und "SCROLL DOWN" bequem durchforsten. Zweimaliges Drücken auf "MODE" verschafft Zutritt zu Block B.

Controller UC-2200

Komplett wird das System allerdings erst durch den "Controller" UC-2200, der die Maße eines Taschenbuches aufweist und über eine große Gummitastatur, ROM-Pack-Schubfach, Uhrenhalterung und einen Thermodrucker (20 Zeichen/sek.) für 58mm Rollenbreite verfügt. Dieses Steuergerät braucht die Uhr nur als Monitor, da das UC-2200 über einen eigenen Prozessor (Z80-kompatibel), ein gutes 26K-Basic (Microsoft) und einen 4K-RAM zum vollwertigen Pocket- bzw. Handheldcomputer taugt. Die Verbindung zur Uhr wird ebenfalls über eine Induktions-

schleife realisiert, so daß Kabelwirrwarr von vorne herein unterbunden bleibt.
Negativ macht sich hier allein die Armbanduhr bemerkbar, die einerseits recht langsam auf Befehlseingaben reagiert, andererseits, wenig Editierkomfort ermöglicht.

Hier würde ich mir eine spürbare Verbesserung in Gestalt eines Extradisplays für das UC-2200 wünschen: eine 20-Zeichen-Anzeige (einzeilig) würde völlig genügen. Dabei sollte die "Kontaktschnittstelle" vorhanden bleiben, da gegen einen Informationsaustausch zwischen UC-2000 (Uhr) und UC-2200 (Steuergerät) natürlich nichts einzuwenden ist.

Bedienungsfreundlichkeit

Bevor ich das Kapitel "Bedienerfreundlichkeit" abhandele, muß ich darauf verweisen, daß der Lieferung der schweizerischen SECOM AG, der ich für die prompte und kostenlose Überlassung einer kompletten Testkonfiguration zu Dank verpflichtet bin, kein Handbuch, "Manual" oder eine Bedienungsanleitung für das UC-2000-System beilag.
Auch die Verpackungseinheiten der drei Geräte waren für die Aufnahme einer Begleitdokumentation nicht vorgesehen.
Ich war also neben einem Quentchen einschlägiger Erfahrung ganz und gar auf das angewiesen, was ich mit dem Begriff "Bedienerfreundlichkeit" versehen habe. Glücklicherweise ist die Anwenderführung im Rahmen des Möglichen als ausgesprochen gelungen zu betrachten.
Daß die Bedienungselemente der Armbanduhr unmißverständlich ausgelegt sind, habe ich bereits vermerkt: geradezu lachen mußte ich jedoch, als ich feststellte, daß die Zusammenarbeit der einzelnen Systemkomponenten über waschechte Menüsteuerung erfolgt. Seiko und Sekom mögen es mir verzeihen: es wirkt aber auch zu komisch, auf dem Miniatur-"Bildschirm" einer Armbanduhr einen perfekten Dialogverkehr zu erleben.
Das Menü weist 4 mögliche Funktionen aus, die durch entsprechende Cursor-Positionierung angewählt werden können. Ist dazu der Wechsel des Betriebsbereiches nötig (z. B. von "Basic" in "APL"=application), so läßt sich das System zur Sicherheit den Wunsch des Benutzers zusätzlich bestätigen. Derlei findet man bei vielen Taschencomputern nicht!
"APL" (Abk. für "Application" ="Anwendung") ruft die im ROM-Pack gespeicherten Sonderleistungen auf. Das Standard-ROM bietet ein kleines Demo-Programm, das die Einsatzbereiche des Systems anschaulich macht, einen Terminkalender sowie zwei Spielprogramme.
Die Spiele "HIT" (ein einfacher "Space-Invader") und "RACE" ("Pferde-Wetten") können einzeln in den Uhren-RAM geladen werden, wobei man allerdings auf die Memo-Blöcke A und B verzichten muß. Dabei ist "RACE" kein übliches Reaktions-, sondern ein Glücksspiel: man erhält zu Beginn ein Einsatzkapital, über das man nach eigenem Ermessen verfügen kann. Die Qualität der Pferde wird anhand einer eingeblendeten Bestenliste beurteilt. Dann wird gestartet, der Rennverlauf angezeigt und die Abrechnung vorgenommen. Spielende ist dann, wenn entweder das Gesamtkapital verspielt oder der Maximalgewinn erreicht ist.

Spiele

Uhrenspiele finden sich in vielen Billigmodellen und stellen nicht unbedingt das dar, was man von einem dreiteiligen "Computer-Uhren-System" erwartet. Dazu kommt, daß ein aus dem ROM überspieltes "Hit" oder "Race" die Speicherfunktionen außer Kraft setzt.
Attraktiver ist da schon der ebenfalls aus dem ROM zu übernehmende "Schedule", ein Terminkalender, der jeweils die kommenden 30 Tage umfaßt. Man gibt einmal das aktuelle Jahr ein: dann werden die Wochentage für den kommenden Monat errechnet und dann 30 "pages" von jeweils 2 Zeilen bereitgestellt, die jeweils mit Datum-Kopfzeilen versehen sind (Beispiel: **01-01 TUE** für Dienstag, 01. Jan. '85) und durchblättert werden können.
Das ergibt eine übersichtliche und einfach zu handhabende Alternative zu den Memos A und B, da automatisch Ordnung in die Notizen gebracht wird und das Datum nicht immer mit gespeichert werden muß.

Fazit

Das UC-2000er System von Seiko ist ein attraktives Spielzeug – das ist nicht zu bestreiten. Bei einem Endverbraucherpreis von etwa 700 Mark (inklusive Mehrwertsteuer) muß man schon ein wenig überlegen, ob für die persönlichen Bedürfnisse nicht unter Umständen ein Taschencomputer günstiger wäre. Wer aber ein Mini-Datensystem wünscht, das ihm ermöglicht, ausgewählte Informationen garantiert bei sich zu haben, das darüber hinaus aber auch die wichtigen Uhrenfunktionen und die Erstellung von Ausdrucken ermöglicht, wird mit dem UC-2000 von Seiko bestimmt keinen Fehlgriff tun. Dazu kommt, daß die Komponenten auch einzeln erhältlich sind, wodurch auch die "kleine Lösung" (Uhr + Tastatur) in Frage kommt.

Wer ständig mit Computern arbeitet, also auch deren Launen und Unzulänglichkeiten kennt, wird vor allem die Präzision der Komponenten zu schätzen wissen: ich habe keinen einzigen Fall erlebt, daß der Drucker mangelhaft transportiert oder die Datenübertragung nicht geklappt hätte.
Gerade bei portablen Geräten ist man auf ein Höchstmaß an Zuverlässigkeit angewiesen. Die Unruhe "funktioniert's nun oder nicht?" kann nervenaufreibender sein als der Verzicht auf elektronische Gedächtnisstützen.
Und hier liegt eine wesentliche Stärke des getesteten Systems.
Die Komponenten sind sehr sorgfältig verarbeitet und hart belastbar.

Nun aber noch einmel tabellarisch die Gegenüberstellung von Vor- und Nachteilen:

Erfreulich:
- Große Vielseitigkeit
- Perfekte Menü-Steuerung
- Großer Speicherplatz
- Gebräuchlicher Prozessor Z-80
- Leistungsstarkes Basic 26K
- Keine Verbindungskabel
- Solide Verarbeitung

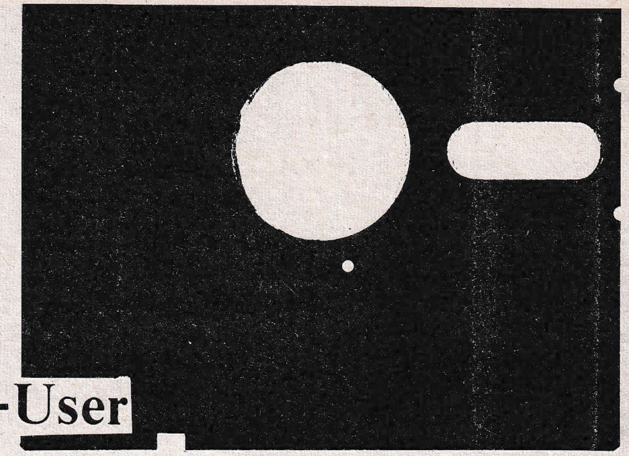
Weniger erfreulich:
- Langsame Uhren-CPU
- Kein Controller-Display

Vertrieb in Deutschland:
Mirwald-Elektronik,
Unterhaching bei München

(sk)

Disketten

schnell wie der Blitz, empfindlich wie rohe Eier! Kleines Einmaleins für Floppy-User



Die große Leserumfrage in CPU hat zwar erbracht, daß durchaus nicht so viele Besitzer eines Heimcomputers mit Disketten-Laufwerken arbeiten – nämlich nicht einmal jeder Dritte: dennoch gehört die Floppy zu denjenigen Peripherien, die man sich am sehnlichsten wünscht. Manch einer verzichtet lieber auf einen Nadeldrucker als auf einen solchen Magnetplatten-Speicher. Die Gründe dafür liegen auf der Hand.
Disketten-Laufwerken sagt man eine gegenüber Daten-Rekordern beachtliche Zugriffsgeschwindigkeit nach. Auch liefern sie praktische Inhaltsverzeichnisse auf den Bildschirm, Directories genannt: man sieht auf einen Blick, was die jeweilige Floppy an Programmen und Daten enthält. Also braucht man ganz einfach eine Disketten-Station!

Was ist eine Diskette

In der kurzen Einleitung habe ich bewußt verschiedene Begriffe für ein und dasselbe parallel benutzt: Diskette, Floppy, Magnetplatte. Dazu kommen noch unterschiedliche hersteller-spezifische Namen wie zum Beispiel 'Flexidisk', 'Quickdisk' und ähnliches mehr, wodurch gerade Einsteiger leicht verwirrt werden können. Deshalb zunächst mal eine grundsätzliche Klärung der einzelnen Begriffe:

Diskette
heißt auf deutsch soviel wie "kleine Scheibe" (vom lateinischen Wort 'discus' = Scheibe)

Floppy
ist ein englisches Eigenschaftswort, das auf deutsch mit 'biegsam' wiedergegeben werden kann.

Magnetplatte
ist in sofern der wohl präziseste Begriff, da er auch das Aufzeichnungsprinzip enthält: im Gegensatz zur Schallplatte nämlich, die Information in Form einer kraxeligen Spiralspur entgält, funktioniert die Magnetplatte grundsätzlich wie ein Tonband – durch Magnetisierung ihrer Oberfläche.

Weil Magnetplatten hauchdünn und dadurch floppy, also biegsam sind, werden sie in eine Karton- oder Kunststoffhülle gepackt, in der sie immer bleiben. Ein Knick in der Scheibe selbst würde nämlich zu Lesefehlern und dadurch zu völligem Programmverlust führen.
Damit man aber trotz der guten Verpackung mit seiner Diskette arbeiten kann, verfügt die Plattenhülle über mehrere Öffnungen.
In der Plattenmitte befindet sich ein großes Loch, das die gleiche Funktion wie die Öffnung in Ihren Schallplatten besitzt: da greift die Antriebsspindel des Laufwerkes hinein, um die Floppy im Griff zu haben. Wenn Sie bei billigen Disketten mit dem Finger am Rand der Öffnung entlang fahren, spüren Sie, wie dünn die eigentliche Scheibe ist. Ich sagte deshalb 'billige' Diskette, weil bei teureren Fabrikaten in der Regel Verstärkungsringe eingebaut sind, die ein "Ausfransen" der Floppy verhindern sollen.

Neben der großen gibt es noch eine länglich-ovale kleine Öffnung, die wie der Querschnitt einer etwa zu groß geratenen Schnupfenkapsel aussieht. Da man in dieser Öffnung die Disketten-Oberfläche ungeschützt sieht, ist besondere Vorsicht am Platze. Niemals mit den Fingern die Magnetplatte berühren!
Die kleine Öffnung ist das Schreib/Lese-Fenster, durch das die Tonköpfe der Diskettenstation an die Floppy gelangen.

Sind alle Disketten gleich?

So, wie es verschiedene Videokassetten oder Filmtypen gibt, so gibt es auch Unterschiede bei Floppy-Disks.
Da ist zuerst einmal die Größe. Diskettenformate werden in Zoll bzw. Inch angegeben, wobei 1 Zoll etwa 2,31 Zentimetern entspricht. Im Handel sind 8"-Disketten ("=Abkürzung für Zoll) für industriellen Einsatz, Floppies mit 5,25 Zoll Durchmesser für die meisten Heim- und Personalcomputer sowie verschiedene Mini-Formate von 3,5" bis hinunter zu 2,8"-Disketten mit Hardcover.
Doch damit der Unterschiede nicht genug. Es gibt einseitig (engl.: single-sided) und doppelseitig (double-sided) nutzbare Disketten. Und darüber hinaus gibt es solche mit Standard-Schreibdichte (single-density) sowie Exemplare, die doppelt so dicht aufzeichnen (double-density).
Was die Qualität des Materials ausmacht, so gibt es sicher auch einzelne Unterschiede. Die deutliche Abgrenzung wie bei Tonbändern jedoch (etwa: Ferro, Chrom, Ferrochrom, Reineisen) gibt es nicht. Das liegt schon daran, daß die Aufzeichnung von Datensätzen nicht die Ansprüche wie eine Musikaufnahme stellt – die Jagd nach hohen und höchsten Frequenzen findet also nicht statt.

Wie behandele ich Disketten?

Wie wir festgestellt haben, ist die Diskette ein empfindliches Speichermedium. Es kann zwar keinen "Bandsalat" wie bei Tonbandkassetten geben, dafür sind aber diverse andere Punkte zu be-

achten. Hier das wichtigste in Kürze:

- **Biegen** oder **knicken** Sie Ihre Disketten n i e m a l s!
- Beschriften Sie Ihre Disketten **mit Faserschreiber** und nie mit Kugelschreiber oder hartem Bleistift!
- Bringen Sie **keine magnetischen Gegenstände in die Nähe** Ihrer Disks!
- **Vermeiden Sie extreme Temperaturen** in der Umgebung Ihrer Disketten!
- **Berühren Sie niemals die Disketten-Oberfläche im Schreib/Lese-Fenster** (ovale Hüllen-Öffnung)!

Derartige Verbote sollen Ihnen nicht den Spaß am Umgang mit dem Disketten-Laufwerk verderben – im Gegenteil! Bedenken Sie, wieviel Mühe die Erstellung des einen oder anderen Programmes gekostet hat. Es wäre doch schade, wenn durch eine dumme Unachtsamkeit wertvolle Daten verloren gingen.

Die gute Behandlung Ihres Diskettenbestandes fängt schon bei der Aufbewahrung an: Floppies mögen es gar nicht, kreuz und quer zwischen anderen wichtigen Dingen am Arbeitsplatz – vom Bedienerhandbuch bis zur Cola-Dose, vom Drucker-Interface bis zur Chipstüte – herumzugammeln. Computerläden sowie die einschlägigen Abteilungen der Warenhäuser sind voll von praktischen Disketten-Archiven, Kunststoff-Boxen mit sinnvoller Einrichtung: Disketten sind, weil dünn und handlich, ideal zum sicheren Verstauen in solch einem Kasten. Wenn man sie auch noch deutlich beschriftet, findet man schnell das gesuchte Programm und spart sich umständliches Durchforsten der Directories.

Von Magnetplatten mit unersetzlichen Informationen und besonders wertvollen Aufzeichnungen sollten geprüfte Kopien erstellt und an sicherem Platz aufbewahrt werden. Darüber hinaus ist es ratsam, solche Scheiben mit Schreibschutz zu versehen.

Fast wie bei Compact-Cassetten: der Disk-Schreibschutz!

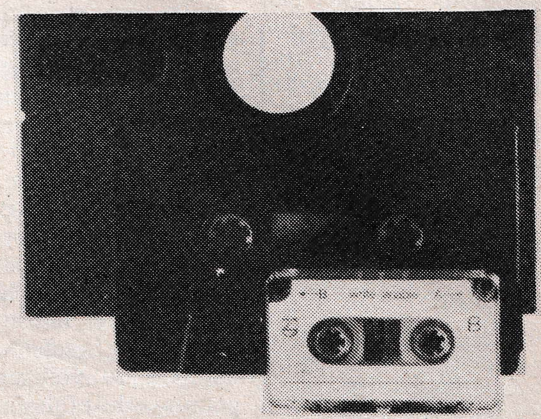
Seit es die Compact-Cassette für Tonbandgeräte gibt, gibt es auch den Aufnahmeschutz: auf der Rückseite jeder Kassette sind zwei Plastikzungen angebracht, die Aufnahmen ermöglichen. Werden sie entfernt, wird die Aufnahmetaste des Rekorders blockiert und ein Überspielen verhindert. Fast genauso ist es bei Disketten: hier allerdings gilt eine Aussparung als Voraussetzung für Beschreibbarkeit (das Musikaufnahmen ohnehin nicht in Frage kommen, spricht man bei Floppies von 'Beschreiben' statt 'Bespielen'). Fehlt die Aussparung, so ist die Diskette geschützt.

Das Beschreiben nicht einfach "Record"

Das erste, was einen frischgebackenen Floppy-Besitzer irritiert, ist das Fehlen von Bedienungselementen, wie sie vom Datenrekorder her bekannt sind. Keine 'Play'- oder 'Record'-Taste, kein schneller Vor- und Rücklauf. Alles muß über spezielle Tastaturbefehle realisiert werden. Verantwortlich dafür ist oft das sogenannte Disk-Operating-System (DOS), das entweder im Disketten-Laufwerk oder im Computer selbst untergebracht ist. Ein solches 'Disketten-Betriebssystem' liefert Ihnen den gesamten Vorrat an Anweisungen, um eine Floppy auf ihren Gebrauch vorzubereiten, sie zu beschreiben, zu lesen oder auch zu kopieren. Ja, Sie haben richtig gelesen: eine Diskette kann nicht so einfach benutzt werden wie eine Tonbandkassette. Vor ihrer Erstbenutzung muß sie 'formatiert' werden, d.h. demjenigen Aufzeichnungsstandard angepaßt werden, den Ihr Computer für seine SAVE-Aktionen wählt.

Sie werden nun sicher wissen wollen, weshalb man nicht einfach drauf los 'SAVEn' kann, wie man es vom Magnetband her gewohnt ist: nun, das ist ganz einfach!

Das Tonband kann sich, egal welcher Computer es beschreibt, jeweils nur in eine Richtung bewegen und somit in einer bestimmten Folge Daten aufzeichnen. Man spricht deshalb auch von 'sequentiellen Dateien' auf dem Magnetband, weil alle Informationen 'hintereinander' aufgeschrieben werden. Dem gegenüber kann die Disketten-Station bestimmte Sektoren auf der Floppy direkt ansteuern, ohne vom Anfang bis zum Ende suchen zu müssen. Das geht natürlich nur dann, wenn auf der Floppy Ordnung herrscht. Damit das technisch auch klappt, gibt es den 'Formatierungslauf', mit dem Ihr Rechner den Speicherplatz



auf der neuen Diskette so aufteilt, wie er das bei allen früheren Disketten auch getan hat. Er bringt ein spezielles System von Spalten und Spuren an, so daß später einzelne Segmente bequem angewählt werden können. Dieses Formatieren ist nicht etwa kompliziert, sondern über einen einfachen Befehl aus dem Sprachumfang des DOS zu starten.

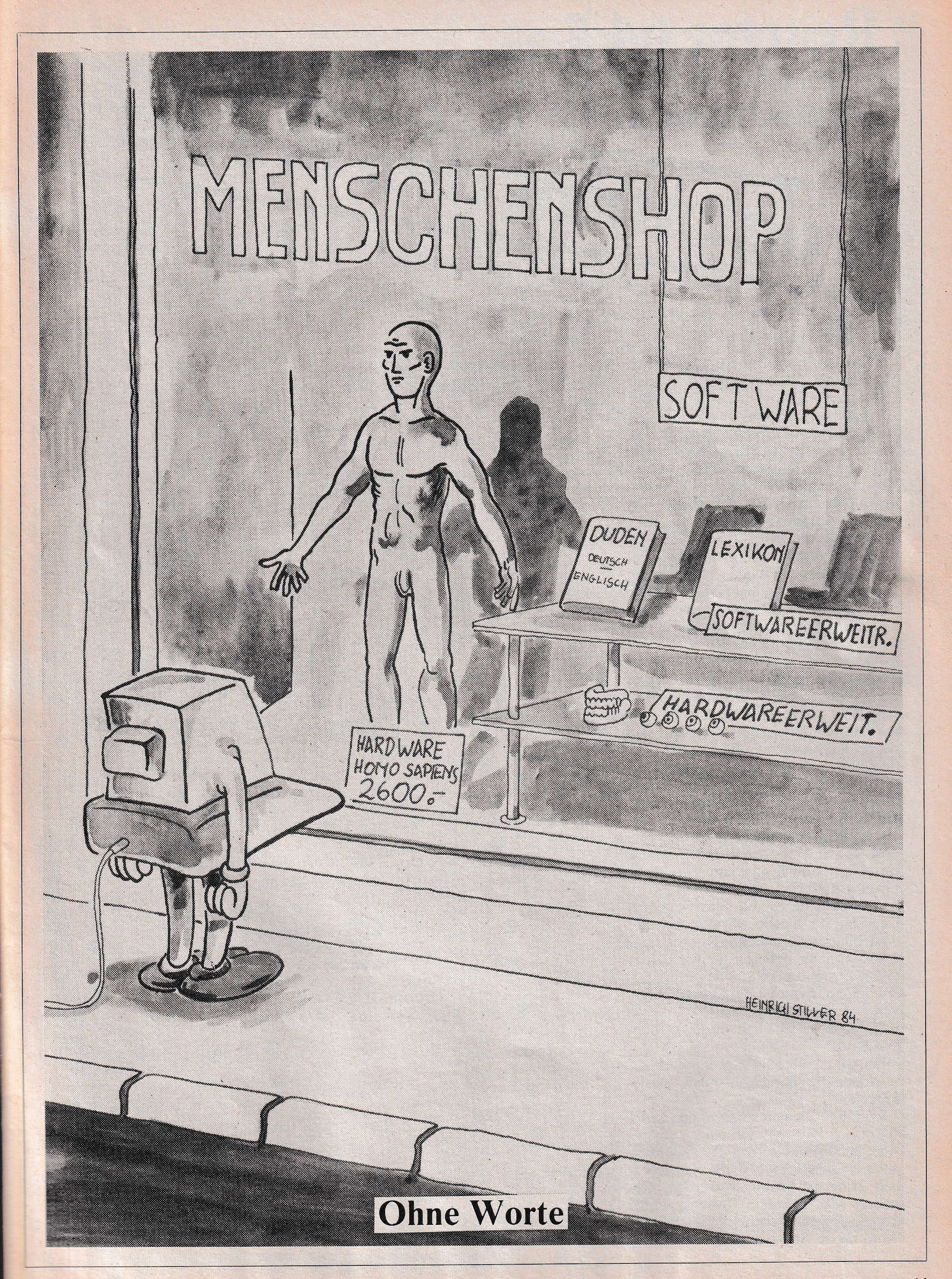
Praktisch: die 'Directory'!

Ist für den Umsteiger von Band auf Disk das Formatieren auch etwas lästig und ungewohnt, so findet er sicher die 'Directory', also das Inhaltsverzeichnis, um so besser. Jetzt entfällt doch endlich das blödsinnige Suchen nach bestimmten Programmen oder Dateien, wie er es bei der Datasette häufig erlebte – zumal Bandzählwerke selten zuverlässig arbeiten.

Die Directory listet sämtliche auf der betreffende Diskette gespeicherten Programme der Reihe nach auf. Man sucht sich das gewünschte Programm aus, schreibt einen Lade-Befehl sowie den Programmnamen auf den Bildschirm (bei manchen DOS-Versionen reicht ein Ansteuern des jeweiligen Programmes mit dem Cursor) – und schon lädt die Floppy das Programm in den Arbeitsspeicher Ihres Computers.

FAZIT:
Disketten-Laufwerke sind praktische Helfer, auch wenn natürlich gilt: die perfekte Lösung gibt's (noch) nicht. So vielversprechend also der eine oder andere Hersteller sein Gerät auch anpreisen mag: man sollte nicht vergessen, daß die Vorzüge nicht in Reinkultur blühen. Angesichts der häufigen Pannen durch falsch justierte Schreib/Lese-Köpfe oder ausgefranste Disketten sollte man die Investition meines Erachtens nur dann in Betracht ziehen, wenn man den Faktor Zeit wirklich ernst nehmen muß und schellsten Zugriff dringend benötigt. Wer den Rechner privat nutzt und sich leisten kann, ein paar Minuten länger zu warten, ist mit dem guten alten Magnetband-Speicher bestens beraten.

(sk)


MENSCHENSHOP
SOFT WARE
DUDEN
DEUTSCH
ENGLISCH
LEXIKON
SOFTWAREERWEITR.
HARDWAREERWEIT.
HARDWARE
HOMO SAPIENS
2600.-
HEINRICH STILLER 84
Ohne Worte

Reactor 64

Commodore 64

Programm des Monats

in Simons-Basic

Ein Atomreaktor ist kurz vor dem Durchgehen. Es gibt nur eine einzige Chance ihn davor zu bewahren! Der Ausweg besteht darin, die Wände des Reaktorkerns so mit Energie zu versorgen, daß sie in der Lage sind den Reaktorkern zu stabilisieren. Dieses ist eine sehr problematische Angelegenheit und deshalb hat man Sie, den größten lebenden Nuklearspezialisten, zu Hilfe geholt. Hierfür stehen Ihnen fünf Sonden zur Verfügung. Sie müssen diese erst mit Energie betanken, die es beim Power-Zeichen gibt. Anschließend müssen Sie mit der Sonde eine Wand des Reaktorkerns vorsichtig berühren. Verfügt die Wand über genug Energie, so verfärbt sie sich weiß. Sind alle vier Reaktorkernwände weiß, haben Sie es mal wieder geschafft. Die Aufgabe hört sich recht leicht an, jedoch explodiert die Sonde immer, falls sie etwas berührt, da sie sehr empfindlich ist. Außerdem schwirren im Kern immer einige Neutronen herum, die sie behindern und ab dem achten Level taucht sogar ein Proton auf.

Haben Sie Ihre Aufgabe bewältigt, werden Sie zu einem neuen Krisenherd gerufen, der noch schwieriger (höherer Level) ist.

Als Hilfe gibt es die Stasis-Energie. Drücken Sie den Feuerknopf, hält auf dem Bildschirm alles an. Nur Sie können sich noch bewegen. Leider dauert dieser Zustand maximal nur einige Sekunden an, dann ist die Stasis-Energie verbraucht.

Punkte gibt es nach jedem überstandenen Level:

- Punkte für übrig gebliebene Zeit
- Punkte für übrig gebliebene Stasis-Energie
- Punkte für den entsprechenden Level

Press SPACE bar

Alle 2000 Punkte gibt es außerdem noch eine Zusatzsonde, die Sie sicherlich dringend benötigen werden. Gesteuert wird mit dem Joystick, wobei der Port Nebensache ist.

Wem kein Joystick zur Verfügung steht, kann es mal mit den Tasten [←], [1], [2], [CTRL], und [SPACE] versuchen.

```
10 REM
20 REM
30 REM      *** REACTOR 64 ***
40 REM
50 REM      A REAL ACTION-GAME
60 REM (C) 1984 BY  LIPPE ENTERPRISES
70 REM      [C.V.D.LIPPE,ASTERNSTR.16]
80 REM      [          1]
90 REM
91 REM WRITTEN IN  SIMON'S BASIC
92 REM         AND  MACHINE-CODE
93 REM              FOR THE CBM-64
94 REM
95 :
96 POKE56,122:CLR:REM BASIC SPEICHER RES
ERVIEREN FUER MC-ROUTINE
97 EXEC MC-ROUTINE
98 :
99 :
100 REM VARIABLEN INITIALISIEREN
110 GAME=$7A00
120 SID=54272
130 HI=0
135 SLEVEL=0
140 :
200 REM VORSPANN
205 EXEC SPRITES
210 EXEC VORSPANN
220 EXEC RESET
250 :
1000 REM MENUE
1004 PRINTCHR$(147):COLOUR15,15
1005 EXEC ZEICHENSATZ
1010 PRINTAT(10,0)"■*** REACTOR 64 ***"
1020 PRINT"■■■■■SELECT MODE"
1030 PRINT"■■■■A■  GAME MODE"
1040 PRINT"■■■■B■  INSTRUCTIONS"
1042 PRINT"■■■■C■  SELECT STARTING LEVE
L"
1045 PRINT"■■■■D■  END"
1050 PRINTAT(11,12)"HIGHSCORE:"HI
1060 FLASH11,10:POKE198,0
1070 PRINTAT(0,19)" ":CENTRE"■■ 1984 B
Y  LIPPE ENTERPRISES":PRINT""
1080 CENTRE"[C.V.D.LIPPE,ASTERNSTR.16,4
98BUENDE1]"
1090 FORZ=1TO5000
1100 GETA$
1110 IFA$="A"THENOFF:GOTO10000
1120 IFA$="B"THENOFF:EXEC ANLEITUNG
1130 IFA$="B"THEN1000
1135 IFA$="C"THENOFF:EXEC SELECT
1136 IFA$="C"THEN1000
1137 IFA$="D"THEN COLD
1140 NEXTZ
1150 GOTO200
1160 :
1170 :
10000 REM SPIELEN
10010 LEVEL=SLEVEL
10015 SPIELER=5
```

```
10020 SC=0:SHIP=1
10030 IFLEVEL>7THENPRO=1:ELSE:PRO=0
10040 EXEC LEVEL
10045 EXEC ZEICHENSATZ
10046 EXEC SPRITES
10050 EXEC BILDAUFBAU
10060 EXEC SOUND2
10070 EXEC SOUND3
10080 EXEC SOUND4
10090 SYS GAME
10095 EXEC RESET
10100 E=PEEK(2)
10110 IFE=8THEN EXEC POINTS
10120 IFE=8THEN LEVEL=LEVEL+1:GOTO10030
10130 IFE=1THEN PRINTAT(8,4)"■CHINA SYN
DROM!":PRINTAT(■8,6)"YOU HAVE LOST!"
10140 RCOMP GOTO 11000
10150 SPIELER=SPIELER-1
10160 IFSPIELER>0THEN10030
10198 :
10199 :
11000 REM ENDE VOM SPIEL
11002 EXEC RESET
11003 RLOCMOB0,234,185,3,0
11004 FILL 8,11,8,8,32,0
11005 MOB SET0,11,7,0,0
11010 FORX=128TO0STEP-1
11020 RLOCMOB0,120+COS(X/5)*X,128+SIN(X
/5)*X,3,0
11030 NEXTX
11050 REM HIGHSCORE
11060 IF SC<=HI THEN11200
11070 PRINTAT(10,20)"■HIGHSCORE"
11080 FLASH9,2
11090 EXEC SOUND2
11100 POKESID+15,156:POKESID+14,72
11110 POKESID+18,65
11120 FORX=0TO255
11130 POKESID+24,XAND15
11170 NEXT:OFF
11180 HI=SC:POKESID+18,0
11200 PAUSE5:EXEC RESET
11210 GOTO1000
11220 :
12000 PROC POINTS
12010 EXEC SOUND2
12020 Z=PEEK($03CE):POKESID+15,(160-Z)/2
:POKESID+18,33
12030 FORX=ZTO159:SC=SC+1:POKESID+15,(16
0-X)/2:POKE$E000+800+X,0
12040 PRINTAT(32,19)"■ ":USE"#####.",
STR$(SC):PRINT"■ "
12050 NEXT
12060 Z=PEEK($03CF):POKESID+15,(103-Z)/1
.5
12070 FORX=ZTO103:SC=SC+1:POKESID+15,(10
3-X)/1.5:POKE$E000+192*8+X,0
12080 PRINTAT(32,19)"■ ":USE"#####.",
STR$(SC):PRINT"■ "
12090 NEXT:POKESID+18,0
12100 PRINTAT(4,5)"■SCORE=SCORE+LEVEL*1
00"
12110 FLASH9,10
12120 IFLEVEL=0THENGOTO12180
12130 FORX=1TOLEVEL
12140 POKESID+15,52:POKESID+18,17
12150 PAUSE1:POKESID+18,0:SC=SC+100
12160 PRINTAT(32,19)"■ ":USE"#####.",
STR$(SC):PRINT"■ "
12170 PAUSE1:NEXT:PAUSE2:OFF
12180 PAUSE2:OFF
12190 IFSHIP*2000>SCTHENEND PROC
12200 IFSPIELER>5THENEND PROC
12210 SHIP=SHIP+1:SPIELER=SPIELER+1
12220 PRINTAT(9,17)"■BONUS  SONDE":POKE
SID+18,65
12230 FORX=0TO100:POKESID+15,X
12240 INV17,9,12,1:NEXT:POKESID+18,0
12250 PAUSE1:END PROC
12260 :
15000 PROC SELECT
15010 COLOUR8,8:PRINT"■SELECT LEVEL US
ING JOYSTICK OR KEYBOARD."
15020 PRINTAT(15,10)"■LEVEL:■"SLEVEL"■■
■■ "
15030 GETB$:JO=PEEK(56320)ANDPEEK(56321)

15040 IFB$>CHR$(47)ANDB$<CHR$(58)THENSLE
VEL=VAL(B$)
15050 IF(JOAND1)=0THENSLEVEL=SLEVEL+1:IF
SLEVEL>9THENSLEVEL=0
15060 IF(JOAND2)=0THENSLEVEL=SLEVEL-1:IF
SLEVEL<0THENSLEVEL=9
15070 IFB$=CHR$(13)OR(JOAND16)=0THENFLAS
H11,1:PAUSE2:OFF:END PROC
15080 GOTO15020
15090 PAUSE10:END PROC
19998 :
19999 :
20000 PROC ANLEITUNG
20010 COLOUR14,14:PRINT"■"TAB(10)"■***
_-|- 64 ***";CHR$(14)
20020 PRINT"■IN ■TOMREAKTOR IST KURZ VO
R DEM -URCH-"
20030 PRINT"■GEHEN. -S GIBT NUR EINE EIN
ZIGE -HANCE"
20040 PRINT"■IHN DAVOR ZU BEWAHREN! -ER
■USWEG BE-"
20050 PRINT"■STEHT DARIN, DIE OAENDE DES
_EACTORKERNS"
20060 PRINT"SO MIT -NERGIE ZU VERSORGEN,
DASS SIE IN"
20070 PRINT"DER LAGE SIND DEN _EAKTORKER
N ZU STABI-"
20080 PRINT"■LISIEREN. -IESES IST EINE S
EHR PROBLEM-"
20090 PRINT"■ATISCHE ■NGELEGENHEIT UND D
ESHALB HAT"
20100 PRINT"■MAN -ICH, DEN GROESSTEN LEB
ENDEN /UKLEAR"
20110 PRINT"■PEZIALISTEN ZU |ILFE GEHOLT
.":EXEC■ TASTE
20120 PRINT"■IR STEHEN FUENF ■ONDEN ZUR
XERFUEGUNG."
20130 PRINT"■U MUSST ■IE ERST MIT -NERG
IE BETANKEN,"
20140 PRINT"■DIE ES BEIM -OWER-■EICHEN G
IBT. ■NSCHL-"
20150 PRINT"■IESSEND MUSST -U MIT DER ■O
NDE EINE"
20160 PRINT"■OAND DES _EAKTORKERNS X-_■
-||,| BERUEH-"
20170 PRINT"REN. XERFUEGT DIE OAND UEBER
GENUG"
20180 PRINT"■NERGIE, SO VERFAERBT SIE S
ICH WEISS."
20190 PRINT"■■IND ALLE VIER _EAKTORKERNW
AENDE WEISS,"
20200 PRINT"■SO HAST -U ES MAL WIEDER GE
SCHAFFT.":EXEC■ TASTE
20210 PRINT"■IESES HOERT SICH RECHT LEI
CHT AN,JEDOCH"
20220 PRINT"EXPLODIERT DIE ■ONDE IMMER,
FALLS SIE"
20230 PRINT"■ETWAS BERUEHRT, DA SIE SEHR
EMPFINDLICH"
20240 PRINT"■IST. ■USSERDEM SCHWIRREN IM
/ERN IMMER"
20250 PRINT"■EINIGE /EUTRONEN HERUM, DIE
-ICH BE-"
20260 PRINT"■HINDERN UND AB DEM ACHTEN L
EVEL TAUCHT"
20270 PRINT"■SOGAR EIN -ROTON AUF."
20280 PRINT"■|AST -U ES ABER DENN TROTZD
EM GESCHAFFT,"
20290 PRINT"DANN WIRST -U ZU EINEM NEUEN
/RISENHERD"
20300 PRINT"■GERUFEN, DER NOCH SCHWIERIG
ER (HOEHERER"
20310 PRINT"■LEVEL) IST.":EXEC TASTE
20320 PRINT"■ALS |ILFE FUER -ICH GIBT ES
DIE ■TASIS-"
20330 PRINT"■NERGIE. -RUECKST -U AUF DE
N _EUERKNOPF,"
20340 PRINT"HAELT AUF DEM |ILDSCHIRM ALL
ES AN,AUSSER"
20350 PRINT"-U SELBST, -U KANNST -ICH NO
CH BEWEGEN."
20360 PRINT"■LEIDER DAUERT DIESER ■USTAN
D MAXIMAL"
20370 PRINT"■NUR EINIGE ■EKUNDEN AN, DAN
N IST DIE"
20380 PRINT"■■TASIS-NERGIE VERBRAUCHT."

20390 PRINT"■UNKTE GIBT ES NACH JEDEM U
EBERSTANDENEN"
```

```
20400 PRINT"LEVEL:"
20410 PRINT"■- ¬UNKTE FUER UEBRIG GEBLIE
BENE ♦EIT"
20420 PRINT"■- ¬UNKTE FUER UEBRIG GEBLIE
BENE ♠TASIS-"
20430 PRINT"  ¬NERGIE"
20440 PRINT"■- ¬UNKTE FUER DEN ENTSPRECH
ENDEN LEVEL":EXEC■ TASTE
20450 PRINT"■LLE 2000 ¬UNKTE GIBT ES AU
SSERDEM NOCH"
20460 PRINT"■EINE ♦USATZSONDE, DIR ¬U SI
CHERLICH"
20470 PRINT"■DRINGEND BENOETIGEN WIRST."

20480 PRINT"■■IESTEUERT WIRD MIT DEM 'OY
STICK, WOBEI"
20490 PRINT"■DER ¬ORT /EBENSACHE IST."
20500 PRINT"■DEM KEIN 'OYSTICK ZUR ×ERFU
EGUNG STEHT."
20510 PRINT"■KANNST ¬U ES JA MAL MIT DEN
 |ASTEN [←],"
20520 PRINT"■[1],[2],[¬|_L] UND [♠¬■-¬]
VERSUCHEN."
20550 PRINT"■■■×IEL |LUECK!!!":EXEC TAS
TE
20560 END PROC
20570 :
29000 PROC TASTE
29010 PRINTAT(10,24)"> PRESS ♠¬■-¬ BAR
<"; :PAUSE2:POKE198,0
29020 GETB$:IFB$<>CHR$(32)THEN29020
29030 END PROC
29998 :
29999 :
30000 PROC SOUND1
30040 POKESID+19,0:POKESID+20,64
30050 POKESID+17,4
30060 POKESID+15,2
30080 POKESID+21,0:POKESID+22,30:POKESID
+23,243:POKESID+24,21
30090 POKESID+18,65
30100 END PROC
30999 :
31000 PROC SOUND2
31005 POKESID+18,0
31010 POKESID+15,52:POKESID+14,39
31020 POKESID+17,5
31030 POKESID+19,0
31040 POKESID+20,240
31050 POKESID+24,5
31070 END PROC
31999 :
32000 PROC SOUND3
32005 POKESID+11,0
32010 POKESID+8,104:POKESID+7,78
32020 POKESID+12,64+4
32030 POKESID+13,0
32040 POKESID+24,5
32090 END PROC
32999 :
33000 PROC SOUND4
33010 POKESID+4,0
33020 POKESID+1,15
33030 POKESID+5,13
33040 POKESID+6,15
33050 END PROC
33998 :
33999 :
40000 PROC BILDAUFBAU
40010 COLOUR0,0 :REM UMRANDUNG
40020 FCOL0,0,31,25,5
40030 POKE$CC00,81
40040 FCHR0,1,28,1,80
40050 FCHR1,0,1,23,82
40060 POKE$CC00+24*40,83
40070 FCHR24,1,28,1,84
40080 POKE$CC00+40*24+29,85
40090 FCHR1,29,1,23,86
40100 POKE$CC00+29,87
40110 MMOB1,120,127,120,127,3,0:REM REA
KTORKERN
40120 MOB SET 1,1,7,0,0
40125 FCOL1,1,30,23,7 :REM REAKTORKERNW
AND
40130 FORX=$CC00+334 TO $CC00+334+39*3 S
TEP39
40140 POKEX,90:POKEX+$0C00,14:NEXT
40150 FORX=$CC00+491TO $CC00+491+41*3 ST
EP41
40160 POKEX,91:POKEX+$0C00,14:NEXT
40170 FORX=$CC00+335 TO $CC00+335+41*3 S
TEP41
40180 POKEX,92:POKEX+$0C00,14:NEXT
40190 FORX= $CC00+498 TO $CC00+498+3*39
STEP39
40200 POKEX,93:POKEX+$0C00,14:NEXT
40250 REM SPRITES FUER EXPLOSION VORBERE
ITEN
40260 FORX=4TO7
40270 MMOBX,0,0,0,0,0,0
40280 MOB SETX,10,1,0,0
40290 NEXT
40300 REM ENERGIE-DEPOT
40310 FCHR2,2,2,2,88
40320 MOB SET 2,4,4,0,0
40330 MMOB 2,40,64,40,64,0,0
40400 REM BASIS+SONDE
40410 FCHR 21,26,2,2,96
40420 MOB SET 0,0,10,0,0
40430 MMOB 0,232,216,232,216,0,0
40500 REM LEVEL SPEZIEFISCHES BILD
40505 GOSUB40510:GOTO40600
40510 CGOTO(LEVELAND7)*100+42000
40600 REM TIME-SAEULE INITIALISIEREN
40610 FCOL4,31,1,20,2
40615 FCOL4,31,1,17,8
40620 FORX=0TO19
40630 POKEX*40+$CC00+191,X+100
40640 NEXT
40700 REM RECHTE BILDSCHIRMSEITE FUELLE
N
40710 PRINT AT(30,1)"■REACTOR 64"
40720 PRINTAT(37,3)"■■■■1■■■3■■4■■■
B■■TTTT■■L■■I■■P■■P■■E■■■";
40730 PRINT"E■■N■■T■■E■■R■■P■■R■■I■■S■■E
■■S"
40740 PRINTAT(30,24)"■TIME■"
40750 PRINTAT(34,4)"■S■■T■■A■■S■■I■■S■■
-■■E■■N■■E■■R■■G■■Y"
40760 PRINTAT(33,18)"■SCORE"
40770 PRINTAT(32,19)" ";:USE"#####.",S
TR$(SC):PRINT"■ "
40780 PRINTAT(33,21)"LEVEL"
40790 PRINTAT(32,22)" ";:USE"#####.",S
TR$(LEVEL):PRINT"■ ■"
40795 IFSPIELER>1THENFILL24,35,SPIELER-
1,1,99,10
40800 REM STASIS-SAEULE INIT.
40810 FCOL4,33,1,13,5
40820 FORX=0TO12
40830 POKE$CC00+193+X*40,X+192
40840 NEXT
40850 REM PROTON INIT.
40860 IFPRO=0THENPOKE$03D0,0:GOTO40900
40865 POKE$03D0,1
40870 MOB SET 3,5,15,0,0
40880 MMOB 3,128,135,128,135,0,0
40900 REM NEUTRONEN
40905 E=(LEVELAND7)/4+2
40906 IFLEVEL>15THENE=E+(LEVELAND240)/16
*2
40910 FORZ=1TOE
40920 X=INT(RND(1)*30)
40930 Y=INT(RND(1)*25)
40940 IFPEEK($CC00+X+Y*40)<>32THEN40920
40950 POKE$CC00+X+Y*40,RND(1)*8+64:NEXTZ

40960 END PROC
40998 :
40999 :
42000 REM LEVEL 0
42010 POKE$CC00+481,96
42020 POKE$CC00+62,96
42050 RETURN
42099 :
42100 REM LEVEL 1
42110 FILL2,14,1,6,89,3
42120 POKE$CC00+62,96
42150 RETURN
42199 :
42200 REM LEVEL 2
42205 FORX=0TO1
42210 FILL10+X*7,X*13+3,1,1,94,3
42220 FILL10+X*7,X*13+4,1,1,95,3
42230 FILL10+X*7,X*13+5,1,1,97,3
42240 FILL10+X*7,X*13+6,1,1,98,3
```

```
42250 NEXT
42290 RETURN
42299 :
42300 REM LEVEL 3
42310 PRINTAT(6,4)"▮NA,WER BIST DENN DU
?"
42320 POKE$CC00+804,89:POKE$CC00+804+$0C
00,3
42330 POKE$CC00+742,89:POKE$CC00+742+$0C
00,3
42340 POKE$CC00+121,96
42390 RETURN
42399 :
42400 REM LEVEL 4
42410 FORX=53067TO52794STEP-39
42420 POKEX,89:POKEX+$0C00,3
42430 NEXT
42440 FILL5,1,5,1,89,10
42450 FILL6,16,1,1,89,10
42490 RETURN
42499 :
42500 REM LEVEL 5
42510 PRINTAT(6,7)"▮H▮A▮L▮L▮O▮E▮L▮L▮O▮U
▮L▮U▮Ö▮!"
42520 POKE$CC00+898,96
42590 RETURN
42599 :
42600 REM LEVEL 6
42610 FILL2,25,1,1,94,3
42620 FILL2,26,1,1,95,3
42630 FILL2,27,1,1,97,3
42640 FILL2,28,1,1,98,3:PRINTAT(5,18)"
DER CBM-64 IST SUPER"
42650 FCHR1,1,3,3,96:POKE$CC00+180,96:F
CHR2,2,2,2,88
42690 RETURN
42699 :
42700 REM LEVEL 7
42710 FILL1,15,1,4,89,5:FILL12,1,5,1,8
9,5
42720 FILL11,24,5,1,89,5:FILL20,14,1,4
,89,5
42790 RETURN
44998 :
44999 :
45000 PROC VORSPANN
45010 HIRES0,0:MULTI1,7,0
45015 CSET1:EXEC CBM-64
45020 TEXT15,10,"▮REACTOR 64",3,6,12
45030 TEXT13,8,"▮REACTOR 64",2,6,12
45060 CIRCLE 73,108,50,24,1
45070 CIRCLE 73,108,14,60,1
45090 FORX=0TO331*8STEP331*8
45100 DESIGN3,$E000+449*8+X
45110 @....
45120 @.CC.
45130 @CCCC
45140 @CCCC
45150 @CCCC
45160 @.CC.
45170 @....
45180 @....
45190 NEXT
45200 TEXT5,182,"▮-  1984 BY LIPPE ▮NTE
RPRISES",3,1,5
45210 CIRCLE 8,185,6,7,3
45220 COLOUR2,2:CSET2:MULTI1,7,0
45230 MMOB 1,146,136,146,136,3,0
45240 MOB SET 1,17,13,0,0
45250 Y=3:EXEC SOUND1
45260 FORX=1TO350
45270 MOB SET 1,16+Y,XAND15,0,0:Y=Y-1:I
FY=0THENY=3
45280 POKESID+14,RND(1)*256
45290 NEXTX
45300 PRINT"▮":POKESID+18,0:MOB OFF1:NR
M
45310 END PROC
45399 :
46000 PROC CBM-64
46010 COLOUR0,0
46020 PRINT"▮▮▮▮▮▮▮"
46030 PRINTTAB(5)"▮▮   ▮ ▮   ▮ ▮  ▮ ▮
 ▮    ▮    ▮ ▮ ▮  ▮ ▮"
46040 PRINTTAB(5)"▮ ▮  ▮ ▮ ▮ ▮ ▮ ▮  ▮ ▮
▮ ▮ ▮    ▮ ▮    ▮ ▮  ▮ ▮"
46050 PRINTTAB(5)"▮ ▮    ▮    ▮ ▮ ▮   ▮
▮ ▮   ▮ ▮    ▮ ▮    ▮"
46060 PRINTTAB(5)"▮ ▮    ▮ ▮  ▮ ▮ ▮ ▮
▮ ▮    ▮ ▮  ▮ ▮    ▮ ▮"
46070 PRINTTAB(5)"▮ ▮  ▮ ▮ ▮ ▮ ▮  ▮ ▮ ▮ ▮
 ▮ ▮    ▮ ▮ ▮ ▮    ▮ ▮"
46080 PRINTTAB(5)"▮    ▮ ▮    ▮ ▮ ▮   ▮
▮     ▮     ▮    ▮ ▮"
46090 CENTRE"▮▮LIPPE ▮NTERPRISES PROUDL
Y PRESENTS:"
46200 END PROC
46298 :
46299 :
47000 PROC LEVEL
47010 HIRES0,0:MULTI2,0,6:COLOUR7,7
47020 X$=RIGHT$(STR$(LEVEL),2)
47030 IFLEVEL<10THENX$=CHR$(48)+RIGHT$(X
$,1)
47040 TEXT61,80,"▮LEVEL",2,1,8
47050 TEXT72,90,"▮"+X$,3,4,10
47060 EXEC SOUND1
47070 POKESID+15,15
47075 FORY=1TO0STEP-1
47080 FORX=59TO0STEP-4
47090 REC60-X,70-(X*1.18),40+2*X,60+2.3
*X,Y
47100 POKESID+15,X/4*Y-(20-X/4)*(Y-1)
47110 NEXTX,Y
47120 EXEC RESET
47130 PRINT"▮":CSET0
47500 END PROC
47510 :
47520 :
48000 PROC RESET
48010 FORX=0TO7:MOB OFFX:MMOBX,0,0,0,0
,0,0:NEXT
48020 FORX=0TO24
48030 POKESID+X,0:NEXT
48040 END PROC
48098 :
48099 :
50000 REM ZEICHENSATZ DEFINIEREN
50010 PROC ZEICHENSATZ
50020 MEM:DESIGN2,$E000:REM COPYRIGHT
-ZEICHEN
50030 @.BBBBB..
50040 @B.....B.
50050 @B.BBB.B.
50060 @B.B...B.
50070 @B.B...B.
50080 @B.BBB.B.
50085 @B.....B.
50090 @.BBBBB..
50100 DESIGN2,$E000+80*8:REM REACTORBEG
RENZUNG 1
50110 @BBBBBBBB
50120 @........
50130 @BBB..BBB
50140 @..B..B..
50150 @..BBBB..
50160 @........
50170 @.B.B.B.B
50180 @BBBBBBBB
50200 DESIGN2,$E000+81*8:REACTORBEGRENZ
UNG 2
50210 @BBBBBBBB
50220 @B.......
50230 @B....BBB
50240 @B....B..
50250 @B....B..
50260 @B.BBBB..
50270 @B.B....B
50280 @B.B....B
50300 DESIGN2,$E000+82*8:REM REACTORBEG
RENZUNG 3
50310 @B.B...BB
50320 @B.B....B
50330 @B.BBB.BB
50340 @B...B..B
50350 @B...B.BB
50360 @B.BBB..B
50370 @B.B...BB
50380 @B.B....B
50400 DESIGN2,$E000+83*8:REM REACTORBEG
RENZUNG 4
50410 @B.B...BB
50420 @B.B.....
50430 @B.BBBB..
50440 @B....B..
50450 @B....B..
```

```
50460 @B....BBB
50470 @B.......
50480 @BBBBBBBB
50500 DESIGN2,$E000+84*8:REM REACTORBEG
RENZUNG 5
50510 @BBBBBBBB
50520 @B.B.B.B.
50530 @........
50540 @..BBBB..
50550 @..B..B..
50560 @BBB..BBB
50570 @........
50580 @BBBBBBBB
50600 DESIGN2,$E000+85*8:REM REACTORBEG
RENZUNG 6
50610 @B....B.B
50620 @B....B.B
50630 @..BBBB.B
50640 @..B....B
50650 @..B....B
50660 @BBB....B
50670 @.......B
50680 @BBBBBBBB
50700 DESIGN2,$E000+86*8:REM REACTORBEG
RENZUNG 7
50710 @B....B.B
50720 @BB...B.B
50730 @B..BBB.B
50740 @BB.B...B
50750 @B..B...B
50760 @BB.BBB.B
50770 @B....B.B
50780 @BB...B.B
50800 DESIGN2,$E000+87*8:REM REACTORBEG
RENZUNG 8
50810 @BBBBBBBB
50820 @.......B
50830 @BBB....B
50840 @..B....B
50850 @..B....B
50860 @..BBBB.B
50870 @.....B.B
50880 @BB...B.B
51000 DESIGN2,$E000+90*8:REM REACTORWAN
D 1
51010 @......BB
51020 @.....BB.
51030 @....BBB.
51040 @...BB.B.
51050 @..BBB.B.
51060 @.BB..B..
51070 @BBBBB.B.
51080 @B......B
51100 DESIGN2,$E000+91*8:REM REACTORWAN
D 2
51110 @B......B
51120 @BBBBB.B.
51130 @.BB..B..
51140 @..BB..B.
51150 @...BB.B.
51160 @....BBB.
51170 @.....BB.
51180 @......BB
51200 DESIGN2,$E000+92*8:REM REACTORWAN
D 3
51210 @BB......
51220 @.BB.....
51230 @.BBB....
51240 @.B.BB...
51250 @.B..BB..
51260 @..B..BB.
51270 @.B.BBBBB
51280 @B......B
51300 DESIGN2,$E000+93*8:REM REACTORWAN
D 4
51310 @B......B
51320 @.B.BBBBB
51330 @..B..BB.
51340 @.B..BB..
51350 @.B.BB...
51360 @.BBB....
51370 @.BB.....
51380 @BB......
51400 DESIGN2,$E000+88*8:REM CHR UNTER
POWER
51410 @........
51420 @........
51430 @........
51440 @........
51450 @........
51460 @........
51470 @........
51480 @........
51500 DESIGN2,$E000+89*8:REM HINDERNIS
IM REACTOR
51510 @...BB...
51520 @..B..B..
51530 @.B..B.B.
51540 @B.BBB..B
51550 @B..BBB.B
51560 @.B.B..B.
51570 @..B..B..
51580 @...BB...
51600 FORX=100TO119:REM TIME SETZEN
51610 DESIGN2,$E000+X*8
51620 @BBBBBBBB
51630 @BBBBBBBB
51640 @BBBBBBBB
51650 @BBBBBBBB
51660 @BBBBBBBB
51670 @BBBBBBBB
51680 @BBBBBBBB
51690 @BBBBBBBB
51695 NEXTX
51700 FORX=192TO192+12:REM STASIS SETZ.
51705 DESIGN2,$E000+X*8
51710 @BBBBBBBB
51720 @BBBBBBBB
51730 @BBBBBBBB
51740 @BBBBBBBB
51750 @BBBBBBBB
51760 @BBBBBBBB
51770 @BBBBBBBB
51780 @BBBBBBBB
51790 NEXTX
51800 DESIGN2,$E000+94*8:REM SCHRIFT 'D
ANGER' 1
51810 @........
51820 @.BBBB..B
51830 @..B.B.B.
51840 @..B.B.BB
51850 @..B.B.B.
51860 @.BBBB.B.
51870 @........
51880 @........
51900 DESIGN2,$E000+95*8:REM SCHRIFT 'D
ANGER' 2
51910 @........
51920 @..B..B.B
51930 @B.BB.B.B
51940 @B.BBBB.B
51950 @B.B.BB.B
51960 @B.B...B.B
51970 @........
51980 @........
52000 DESIGN2,$E000+97*8:REM SCHRIFT 'D
ANGER' 3
52010 @........
52020 @BBB.BBB.
52030 @....B...
52040 @.BB.BB..
52050 @..B.B...
52060 @BBB.BBB.
52070 @........
52080 @........
52100 DESIGN2,$E000+98*8:REM SCHRIFT 'D
ANGER' 4
52110 @........
52120 @BBB..B..
52130 @B.B..B..
52140 @BBB..B..
52150 @BB......
52160 @B.B..B..
52170 @........
52180 @........
52200 DESIGN2,$E000+99*8:REM SYMBOL FUE
R ANZAHL SONDEN
52210 @...BBB..
52220 @...B.B..
52230 @.BBBBBBB
52240 @.B.B.B.B
52250 @.BBBBBBB
52260 @...B.B..
52270 @...BBB..
52280 @........
60000 FORX=0TO64STEP64
```

```
60010 DESIGN2,$E000+64*8+X:REM NEUTRON
1
60020 @........
60030 @..BBB...
60040 @.BBBBB..
60050 @.BB.BB..
60060 @.BBBBB..
60070 @..BBB.B.
60080 @.....BB.
60090 @.......B
60100 DESIGN2,$E000+65*8+X:REM NEUTRON
2
60130 @..BBB...
60140 @.BBBBB..
60150 @.BB.BB..
60160 @.BBBBB..
60170 @..BBB...
60180 @..B.B...
60190 @...B....
60195 @...B....
60200 DESIGN2,$E000+66*8+X:REM NEUTRON
3
60220 @........
60230 @...BBB..
60240 @..BBBBB.
60250 @..BB.BB.
60260 @..BBBBB.
60270 @.B.BBB..
60280 @.BB.....
60290 @B.......
60300 DESIGN2,$E000+67*8+X:REM NEUTRON
4
60310 @........
60320 @.BBB....
60330 @BBBBBB..
60340 @BB.BB.BB
60350 @BBBBBB..
60360 @.BBB....
60370 @........
60380 @........
60400 DESIGN2,$E000+68*8+X:REM NEUTRON
5
60410 @........
60420 @....BBB.
60430 @..BBBBBB
60440 @BB.BB.BB
60450 @..BBBBBB
60460 @....BBB.
60470 @........
60480 @........
60500 DESIGN2,$E000+69*8+X:REM NEUTRON
6
60510 @.......B
60520 @.....BB.
60530 @..BBB.B.
60540 @.BBBBB..
60550 @.BB.BB..
60560 @.BBBBB..
60570 @..BBB...
60580 @........
60600 DESIGN2,$E000+70*8+X:REM NEUTRON
7
60605 @...B....
60610 @...B....
60620 @..B.B...
60630 @..BBB...
60640 @.BBBBB..
60650 @.BB.BB..
60660 @.BBBBB..
60670 @..BBB...
60700 DESIGN2,$E000+71*8+X:REM NEUTRON
8
60710 @B.......
60720 @.BB.....
60730 @.B.BBB..
60740 @..BBBBB.
60750 @..BB.BB.
60760 @..BBBBB.
60770 @...BBB..
60780 @........
60790 NEXTX:END PROC
60799 :
60800 PROC SPRITES
60802 DESIGN 0,49152:REM SONDE
60805 @........................
60810 @........................
60820 @.....BBBBBBB............
60830 @.....BBB.BBB............
60840 @.....BB...BB............
60850 @.....B.....B............
60860 @.BBBBBBBBBBBBBBB........
60870 @.BBB.B.....B.BBB........
60880 @.BB..B.BB..B..BB........
60890 @.B...B.B.B.B...B........
60900 @.BB..B..BB.B..BB........
60910 @.BBB.B.....B.BBB........
60920 @.BBBBBBBBBBBBBBB........
60930 @.....B.....B............
60940 @.....BB...BB............
60950 @.....BBB.BBB............
60960 @.....BBBBBBB............
60970 @........................
60980 @........................
60990 @........................
60995 @........................
60999 FORX=0TO1024STEP1024
61000 DESIGN 0,49152+64+X:REM REACTORKE
RN 1
61010 @........BBBBBBBB........
61020 @......BB........BB......
61030 @.....B...BBBBBB...B.....
61040 @....B...BBBBBBBB...B....
61050 @...B...BB......BB...B...
61060 @...B..BB..BBBB..BB..B...
61070 @..B..BB..BBBBBB..BB..B..
61080 @..B.BB..BB....BB..BB.B..
61090 @..B.BB.BB..BB..BB.BB.B..
61100 @..B.BB.BB.BBBB.BB.BB.B..
61110 @..B.BB.BB.BBBB.BB.BB.B..
61120 @..B.BB.BB..BB..BB.BB.B..
61130 @..B.BB..BB....BB..BB.B..
61140 @..B..BB..BBBBBB..BB..B..
61150 @...B..BB..BBBB..BB..B...
61160 @...B...BB......BB...B...
61170 @....B...BBBBBBBB...B....
61180 @.....B...BBBBBB...B.....
61190 @......BB........BB......
61200 @........BBBBBBBB........
61210 @........................
61300 DESIGN 0,49152+64*2+X:REM REACTOR
KERN 2
61310 @.........BBBBBB.........
61320 @.......BBB....BBB.......
61330 @......BB........BB......
61340 @.....B...BBBBBB...B.....
61350 @....B...BBBBBBBB...B....
61360 @...BB..BB......BB..BB...
61370 @...B..BB..BBBB..BB..B...
61380 @..BB.BB..BB..BB..BB.BB..
61390 @..B..BB.BB....BB.BB..B..
61400 @..B..BB.B..BB..B.BB..B..
61410 @..B..BB.B..BB..B.BB..B..
61420 @..B..BB.BB....BB.BB..B..
61430 @..BB.BB..BB..BB..BB.BB..
61440 @...B..BB..BBBB..BB..B...
61450 @...BB..BB......BB..BB...
61460 @....B...BBBBBBBB...B....
61470 @.....B...BBBBBB...B.....
61480 @......BB........BB......
61490 @.......BBB....BBB.......
61500 @.........BBBBBB.........
61510 @........................
61600 DESIGN 0,49152+64*3+X:REM REACTOR
KERN 3
61610 @........................
61620 @.........BBBBBB.........
61630 @....B..BBBBBBBBBB..B....
61640 @...B..BB........BB..B...
61650 @.....BB..BBBBBB..BB.....
61660 @....BB..BBBBBBBB..BB....
61670 @....B..BB......BB..B....
61680 @...BB.BB..BBBB..BB.BB...
61690 @...BB.BB.BB..BB.BB.BB...
61700 @...BB.BB.B....B.BB.BB...
61710 @...BB.BB.B....B.BB.BB...
61720 @...BB.BB.BB..BB.BB.BB...
61730 @...BB.BB..BBBB..BB.BB...
61740 @....B..BB......BB..B....
61750 @....BB..BBBBBBBB..BB....
61760 @.....BB..BBBBBB..BB.....
61770 @...B..BB........BB..B...
61780 @....B..BBBBBBBBBB..B....
61790 @.........BBBBBB.........
61800 @........................
61810 @........................
61815 NEXTX
62000 DESIGN 0,49152+4*64:REM POWER-ZEI
CHEN
```

```
62010 @........................
62020 @........................
62030 @........................
62040 @.BBB.BBB.B...B..........
62050 @.B.B.B.B.B...B..........
62060 @.BBB.B.B.B.B.B.BB.......
62070 @.B...B.B.B.B.B..........
62080 @.B...BBB.BBBBB.B........
62090 @..............B.........
62100 @.............B..........
62110 @...BBB.BBB..B...........
62120 @...B...B.B.BBBBBB.......
62130 @...BB..BBB.....B........
62140 @...B...BB.....B.........
62150 @...BBB.B.B...B..........
62160 @............B...........
62170 @........................
62180 @........................
62190 @........................
62200 @........................
62210 @........................
62300 DESIGN0,49152+5*64:REM PROTON 1
62310 @......BBBB..............
62320 @........................
62330 @........................
62340 @......BBBB..............
62350 @.....B.BB.B.............
62360 @....BB.BB.BB............
62370 @B..B...BB...B..B........
62380 @B..BBBBBBBBBB..B........
62390 @B..BBBBBBBBBB..B........
62400 @B..B...BB...B..B........
62410 @....BB.BB.BB............
62420 @.....B.BB.B.............
62430 @......BBBB..............
62440 @........................
62450 @........................
62460 @......BBBB..............
62470 @........................
62480 @........................
62490 @........................
62500 @........................
62510 @........................
62600 DESIGN0,49152+6*64:REM PROTON 2
62610 @..........BB............
62620 @............BB..........
62630 @.B......................
62640 @.B....BBBB..............
62650 @B....BB..BB.............
62660 @B...BBB..BBB............
62670 @...BBBB..BBBB...........
62680 @...B...BB...B...........
62690 @...B...BB...B...........
62700 @...BBBB..BBBB...........
62710 @....BBB..BBB...B........
62720 @.....BB..BB....B........
62730 @......BBBB....B.........
62740 @..............B.........
62750 @..BB....................
62760 @....BB..................
62770 @........................
62780 @........................
62790 @........................
62800 @........................
62810 @........................
63000 DESIGN0,49152+7*64:REM PROTON 3
63010 @....BB..................
63020 @..BB....................
63030 @..............B.........
63040 @..............B.........
63050 @......BBBB.....B........
63060 @.....BBBBBB....B........
63070 @....BB.BB.BB............
63080 @....BBB..BBB............
63090 @....BBB..BBB............
63100 @....BB.BB.BB............
63110 @B....BBBBBB.............
63120 @B.....BBBB..............
63130 @.B......................
63140 @.B......................
63150 @............BB..........
63160 @..........BB............
63170 @........................
63180 @........................
63190 @........................
63200 @........................
63210 @........................
63300 DESIGN0,49152+8*64:REM EXPLOSION
1
63301 @.....BB.B...............
63302 @....B...B.B.B...........
63303 @...B.BB.BB.B.BB.........
63304 @..BB.B...BB...B.........
63305 @.BBBB..BB..BBB.B........
63306 @.B...BB...BB...B........
63307 @B...BB.BB..BBB..........
63308 @.BB..BBB..B...B.........
63309 @...B.B.....BB...........
63310 @BB.BB.BBBBBB.BB.........
63311 @..B..BBB.....BB.........
63312 @...B....B...B.B.........
63313 @.BBB..BBBB.BB..B........
63314 @...BBB...B..BBB.........
63315 @..BB.B.B.B.BB...........
63316 @......BBB..B............
63317 @........................
63318 @........................
63319 @........................
63320 @........................
63321 @........................
63350 DESIGN0,49152+9*64:REM EXPLOSION
2
63351 @........................
63352 @.....BB...BB............
63353 @...B.....BB.............
63354 @..BB.BB.....BB..........
63355 @.......BB.BB.B..........
63356 @.B.BB...................
63357 @BB...BBB..BB............
63358 @...B......B..B..........
63359 @..BB.BB.BBB.BB..........
63360 @.B......................
63361 @..BB.B.....BB...........
63362 @B.B..B.BB...............
63363 @BB.......B....B.........
63364 @..BB...B..BB.B..........
63365 @.....B.BB...............
63366 @.....BB...B.............
63367 @........................
63368 @........................
63369 @........................
63370 @........................
63371 @........................
63400 DESIGN0,49152+10*64:REM EXPLOSION
 3
63401 @........................
63402 @........................
63403 @......BB....BB..........
63404 @......BB....BB..........
63405 @..BB.......B............
63406 @..BB....................
63407 @......BBB...............
63408 @......BB...B............
63409 @BB........BB............
63410 @BB..BB....B...B.........
63411 @....BB.......BB.........
63412 @........................
63413 @......BB..BB............
63414 @..B...BB..BB............
63415 @..BBB......B............
63416 @........................
63417 @........................
63418 @........................
63419 @........................
63420 @........................
63421 @........................
63500 DESIGN0,49152+11*64:REM SCHRIFT '
GAME OVER'
63501 @.BBB....B...BBB.BBB.BBBB
63502 @B...B..B.B..B.B.B.B.B...
63503 @B......B.B..B.BBB.B.BBB.
63504 @B.BBB.BBBBB.B.....B.B...
63505 @B...B.B...B.B.....B.B...
63506 @.BBB..B...B.B.....B.BBBB
63507 @........................
63509 @........................
63510 @.BBB..B...B.BBBB.BBB...B
63511 @B...B.B...B.B....B..B..B
63512 @B...B.B...B.B....B..B..B
63513 @B...B..B.B..BBB..BBB...B
63514 @B...B..B.B..B....BB....B
63515 @B...B..B.B..B....B.B....
63516 @.BBB....B...BBBB.B..B..B
63517 @........................
63518 @.BBB.BBB.BBB.BBB.B.B.B..
63519 @.B...B.B.B.B.B.B.B.B.B..
```

```
63520 @.BBB.B.B.BBB.BBB.BBB.B..
63521 @...B.B.B.BB..BB...B.....
63522 @.BBB.BBB.B.B.B.B..B..B..
63523 END PROC
63524 :
63525 :
63600 PROC MC-ROUTINE
63602 COLOUR1,1:PRINTCHR$(147)
63604 PRINTAT(5,13)"■READING MACHINE-PR
OGRAMM !"
63606 PRINTAT(4,15)"PLEASE WAIT ABOUT 2
 MINUTES ..."
63610 RESET 63800
63620 SUMME=0:NUMMER=0:GAME=$7A00
63630 READX:IFX>-1THENPOKEGAME+NUMMER,X:
SUMME=SUMME+X:NUMMER=NUMMER+1:GOTO63630
63640 IF SUMME=178325 AND NUMMER=1449 TH
EN END PROC
63650 PRINT"■■ERROR IN DATAS...."
63660 END
63670 :
63800 REM MACHINE-CODE
63801 DATA169,0,133,2,141,205,3,141,206,
3,141,207,3,141,209,3,141,210,3,141
63802 DATA212,3,32,112,123,144,9,32,57,1
22,32,96,124,32,183,124,32,246,123
63803 DATA32,79,124,32,50,125,32,62,125,
165,2,201,0,240,224,76,96,125,169,0
63804 DATA133,251,169,204,133,252,165,25
2,133,254,162,0,161,251,201,64,240
63805 DATA5,176,3,76,44,123,161,251,201,
80,144,3,76,44,123,201,76,144,8,56
63806 DATA233,8,129,251,76,44,123,169,64
,193,251,208,17,56,165,251,233,41,133
63807 DATA253,176,2,198,254,32,69,123,76
,44,123,169,65,193,251,208,17,56,165
63808 DATA251,233,40,133,253,176,2,198,2
54,32,69,123,76,44,123,169,66,193,251
63809 DATA208,17,56,165,251,233,39,133,2
53,176,2,198,254,32,69,123,76,44,123
63810 DATA169,67,193,251,208,17,56,165,2
51,233,1,133,253,176,2,198,254,32,69
63811 DATA123,76,44,123,169,68,193,251,2
08,21,169,76,129,251,24,165,251,105
63812 DATA1,133,253,144,2,230,254,32,69,
123,76,44,123,169,69,193,251,208,21
63813 DATA169,77,129,251,24,165,251,105,
39,133,253,144,2,230,254,32,69,123
63814 DATA76,44,123,169,70,193,251,208,2
1,169,78,129,251,24,165,251,105,40
63815 DATA133,253,144,2,230,254,32,69,12
3,76,44,123,169,71,193,251,208,18,169
63816 DATA79,129,251,24,165,251,105,41,1
33,253,144,2,230,254,32,69,123,230
63817 DATA251,208,2,230,252,165,252,201,
207,240,3,76,65,122,165,251,201,231
63818 DATA240,3,76,65,122,96,161,253,201
,32,240,28,160,0,140,11,212,160,33
63819 DATA140,11,212,161,251,24,105,1,20
1,72,208,2,169,64,201,80,240,250,129
63820 DATA251,96,161,251,129,253,169,32,
129,251,96,173,0,220,45,1,220,133,251
63821 DATA41,1,201,1,240,9,173,1,208,56,
233,8,141,1,208,165,251,41,2,201,2
63822 DATA240,9,173,1,208,24,105,8,141,1
,208,165,251,41,4,201,4,240,9,173,0
63823 DATA208,56,233,8,141,0,208,165,251
,41,8,201,8,240,9,173,0,208,24,105
63824 DATA8,141,0,208,165,251,41,16,201,
16,208,2,56,96,169,0,133,251,169,230
63825 DATA133,252,172,207,3,192,104,208,
2,56,96,169,0,145,251,200,145,251,200
63826 DATA145,251,200,145,251,200,140,20
7,3,169,0,160,0,200,208,253,24,105
63827 DATA1,201,40,208,246,24,96,169,204
,133,252,173,0,208,56,233,24,74,74
63828 DATA74,133,251,173,1,208,56,233,48
,74,74,74,168,192,0,240,15,136,165
63829 DATA251,24,105,40,133,251,144,242,
230,252,76,15,124,161,251,32,61,124
63830 DATA160,1,177,251,32,61,124,160,40
,177,251,32,61,124,160,41,177,251,32
63831 DATA61,124,96,201,32,240,13,201,96
,240,9,141,60,3,165,2,9,2,133,2,96
63832 DATA238,40,208,172,249,207,136,192
,0,208,2,160,3,140,249,207,96,32,148
63833 DATA124,169,32,133,251,169,227,133
,252,172,205,3,200,192,5,240,4,140
63834 DATA205,3,96,160,0,140,205,3,172,2
06,3,169,0,145,251,200,140,206,3,192
63835 DATA159,208,30,165,2,9,1,133,2,76,
169,124,172,206,3,192,135,176,3,240
63836 DATA1,96,169,52,141,15,212,169,65,
141,18,212,96,173,212,3,201,1,208,1
63837 DATA96,169,0,141,18,212,96,173,208
,3,201,1,240,1,96,173,6,208,205,0,208
63838 DATA240,17,144,9,56,233,2,141,6,20
8,76,216,124,24,105,2,141,6,208,173
63839 DATA7,208,205,1,208,240,17,144,9,5
6,233,2,141,7,208,76,241,124,24,105
63840 DATA2,141,7,208,172,251,207,200,19
2,8,208,2,160,5,140,251,207,238,36
63841 DATA208,173,6,208,74,74,74,74,133,
251,173,0,208,74,74,74,74,197,251,240
63842 DATA1,96,173,7,208,74,74,74,74,133
,251,173,1,208,74,74,74,74,197,251
63843 DATA240,1,96,165,2,9,4,133,2,96,17
3,209,3,201,1,240,1,96,238,39,208,96
63844 DATA173,212,3,201,1,240,1,96,238,2
11,3,238,211,3,173,211,3,201,96,240
63845 DATA4,141,15,212,96,169,0,141,212,
3,141,18,212,96,165,2,201,2,240,3,76
63846 DATA160,126,173,60,3,201,88,208,12
,169,1,141,209,3,169,0,133,2,76,22
63847 DATA122,173,209,3,201,1,240,3,76,1
60,126,173,60,3,201,90,208,54,169,1
63848 DATA141,78,217,141,117,217,141,156
,217,141,195,217,173,0,208,56,233,8
63849 DATA141,0,208,173,1,208,56,201,91,
141,1,208,173,210,3,9,1,141,210,3,201
63850 DATA15,208,7,169,8,133,2,76,160,12
6,76,125,126,173,60,3,201,91,208,54
63851 DATA169,1,141,235,217,141,20,218,1
41,61,218,141,102,218,173,0,208,56
63852 DATA233,8,141,0,208,173,1,208,24,1
05,8,141,1,208,173,210,3,9,2,141,210
63853 DATA3,201,15,208,7,169,8,133,2,76,
160,126,76,125,126,173,60,3,201,92
63854 DATA208,54,169,1,141,79,217,141,12
0,217,141,161,217,141,202,217,173,0
63855 DATA208,24,105,8,141,0,208,173,1,2
08,56,233,8,141,1,208,173,210,3,9,4
63856 DATA141,210,3,201,15,208,7,169,8,1
33,2,76,160,126,76,125,126,173,60,3
63857 DATA201,93,208,54,169,1,141,242,21
7,141,25,218,141,64,218,141,103,218
63858 DATA173,0,208,24,105,8,141,0,208,1
73,1,208,24,105,8,141,1,208,173,210
63859 DATA3,9,8,141,210,3,201,15,208,7,1
69,8,133,2,76,160,126,76,125,126,76
63860 DATA160,126,169,0,141,15,212,141,2
11,3,169,65,141,18,212,169,1,141,212
63861 DATA3,169,0,133,2,169,0,141,209,3,
169,10,141,39,208,76,22,122,165,2,201
63862 DATA8,208,1,96,201,1,208,3,76,82,1
27,169,129,141,4,212,169,12,141,24
63863 DATA212,169,0,141,11,212,141,18,21
2,173,0,208,141,8,208,141,10,208,141
63864 DATA12,208,141,14,208,173,1,208,14
1,9,208,141,11,208,141,13,208,141,15
63865 DATA208,169,8,141,248,207,160,0,23
8,39,208,206,8,208,206,9,208,238,10
63866 DATA208,206,11,208,206,12,208,238,
13,208,238,14,208,238,15,208,173,17
63867 DATA208,233,1,41,7,9,152,141,17,20
8,173,22,208,105,1,41,7,9,200,141,22
63868 DATA208,192,40,208,3,238,248,207,1
92,80,208,3,238,248,207,169,16,162
63869 DATA0,232,234,208,252,56,233,1,201
,0,208,243,200,192,128,208,171,169
63870 DATA155,141,17,208,169,200,141,22,
208,169,0,141,4,212,173,21,208,41,15
63871 DATA141,21,208,96,169,129,141,4,21
2,169,15,141,24,212,169,17,141,18,212
63872 DATA141,11,212,160,5,169,0,162,0,2
38,32,208,238,33,208,238,8,212,238
63873 DATA15,212,232,208,241,142,11,212,
162,17,142,11,212,24,105,1,201,255
63874 DATA208,224,169,0,141,4,212,169,12
9,141,4,212,136,192,0,208,207,169,0
63875 DATA141,32,208,141,33,208,141,11,2
12,141,4,212,141,18,212,96,-1

READY.
```

Concentration

Commodore 64



Ein sehr bekanntes Spiel wurde hier in Basic umgesetzt

Das Spielfeld besteht aus 28 Feldern. 26 davon sind mit dem gesamten Alphabet gefüllt. Da es grafisch nicht anders aufgegangen ist, wurde der 27. Buchstabe durch einen „*" ersetzt. Auch dieser belegt ein Feld. Das 28. Feld bleibt leer, damit man die Zeichen untereinander vertauschen kann. Die an das Leerfeld angrenzenden Zeichen können mit den Tasten A = links, S = rechts, W = oben, Z = unten mit dem Leerfeld vertauscht werden. Die Zeichen müssen nun von links oben beginnend in alphabetischer Folge gereiht werden. (1. Zeile A/B/C/D, 2. Zeile E/F/G/H, ... 7. Zeile Y/Z/*/leer).

Ziel des Spieles ist es nun, mit möglichst wenig Versuchen das Spielfeld zu ordnen. (Der Computer mischt gut.) Es sieht einfacher aus als es ist.

(Der Rekord liegt bei 369 Versuchen!) Ist das nicht ein Anreiz? Probieren Sie's!

```
10 REM*******************
20 REM*  CONCENTRATION   *
30 REM*  1984  MR-SOFT   *
40 REM*-----------------*
50 REM*                 *
60 REM*                 *
70 REM*-----------------*
80 REM* MADE IN AUSTRIA *
90 REM*******************
100 DATA162,0,142,14,220,162,51,134,1,169,0,133,252,169,56,133,253,169,0
110 DATA133,254,169,208,133,255,162,0,160,0,177,254,145,252,200,208,249,23
120 DATA224,16,240,7,230,253,230,255,76,29,192,162,55,134,1,162,1,142,14
130 DATA220,169,0,133,252,169,160,133,253,162,0,160,0,177,252,145,252,200
140 DATA208,249,232,224,32,240,5,230,253,76,69,192,198,1,162,83,142,190,16
150 DATA162,111,142,28,160,162,192,142,29,160,162,3,142,32,208,142,33,208
160 DATA96,32,158,183,138,72,32,253,174,32,158,183,104,168,24,32,240,255
170 DATA32,253,174,76,164,170
180 FORI=49152TO49286:READA:POKEI,A:NEXT
190 SYS49152:V=53248:POKEV+24,31:RE=2000
200 S=54272:FORI=0TO24:POKES+I,0:NEXT
210 POKES+24,15:POKES+1,20:POKES+5,15:POKES+6,5
220 POKE56,56:PRINT"♥":GOSUB40000:GOTO30000
10000 REM****************
10010 REM*SPIELABLAUF    *
10020 REM****************
10030 OK=0:VE=0:GOSUB35000
10035 LET17,1,"█RICHTIGE    :"OK
10040 LET17,5,"TAGESREKORD:"RE
10050 LET17,9,"VERSUCHE    :"VE
10060 K$="ICH MISCHE":GOSUB42000
10070 GOSUB41000
10080 K$="LOS GEHT'S":GOSUB42000
10090 GOSUB36000
10100 K$="RICHTUNG WAEHLEN":GOSUB42000
```

```
10110 GETA$
10120 IFA$="W"ANDPO<5THEN10500
10130 IFA$="Z"ANDPO>24THEN10500
10140 IFA$="A"AND(PO=1OR(PO-1)/4=INT((PO-1)/4))THEN10500
10150 IFA$="S"AND(PO=28ORPO/4=INT(PO/4))THEN10500
10160 IFA$="H"THENGOSUB43000:GOTO10100
10170 IFA$="W"THEN10600
10180 IFA$="A"THEN10700
10190 IFA$="S"THEN10800
10200 IFA$="Z"THEN10900
10210 IFA$="*"THEN12050
10220 GOTO10110
10500 POKE198,0:K$="NICHT MOEGLICH  ":GOSUB42000
10510 FORI=1TO1000:NEXT
10520 POKE198,0:GOTO10100
10600 C$(PO)=C$(PO-4):PO=PO-4:VE=VE+1
10610 C$(PO)=B$(28):GOTO11000
10700 C$(PO)=C$(PO-1):PO=PO-1:VE=VE+1
10710 C$(PO)=B$(28):GOTO11000
10800 C$(PO)=C$(PO+1):PO=PO+1:VE=VE+1
10810 C$(PO)=B$(28):GOTO11000
10900 C$(PO)=C$(PO+4):PO=PO+4:VE=VE+1
10910 C$(PO)=B$(28):GOTO11000
11000 REM*****************
11010 REM*TEST OB OK     *
11020 REM*****************
11025 POKES+4,33:POKES+4,32
11030 GOSUB36000:FORI=1TO28
11040 IFB$(I)=C$(I)THENOK=OK+1
11050 NEXT:LET29,9,"■"VE
11055 LET29,1,"■"OK"▮ ":POKE198,0
11060 IFOK=28THENOK=0:GOTO12000
11070 OK=0:GOTO10110
12000 REM*****************
12010 REM*ALLES RICHTIG *
12020 REM*****************
12030 K$="ICH GRATULIERE !  ":GOSUB42000
12040 FORI=1TO2000:NEXT:IFRE>VETHENRE=VE
12050 K$="NOCH EINMAL ? (J/N) ":GOSUB42000
12060 GETA$:IFA$<>"J"ANDA$<>"N"THEN12060
12070 IFA$="N"THEN13000
12080 K$="OK! MOMENT BITTE     ":GOSUB42000
12090 FORI=1TO1000:NEXT
12100 GOTO10000
13000 REM*****************
13010 REM*SPIELENDE      *
13020 REM*****************
13030 K$="BIS ZUM NAECHSTENMAL":GOSUB42000
13040 FORI=1TO25STEP4
13050 IF(I+7)/8<>INT((I+7)/8)THEN13080
13060 FORJ=0TO3:C$(I+J)=B$(27):NEXTJ:GOTO13100
13070 C$(I)=B$(3):C$(I+1)=B$(9)
13080 C$(I)=B$(3):C$(I+1)=B$(9)
13090 C$(I+2)=B$(1):C$(I+3)=B$(15)
13100 NEXT
13110 GOSUB36000
13120 POKE198,0:WAIT198,1:SYS58648:END
30000 REM*****************
30010 REM*NEUE ZEICHEN   *
30020 REM*****************
30030 READB
30040 A=PEEK(63):IFA=200THEN10000
30050 FORI=0TO7:POKE14336+I+A*8,B
30060 READB:Q=Q+B:NEXT:GOTO30040
32076 DATA95,95,95,95,95,96,63,0
32079 DATA0,63,96,95,95,95,95,95
32080 DATA0,252,6,250,250,250,250,250
32099 DATA0,255,0,255,255,255,255,255
32100 DATA255,255,255,255,255,0,255,0
32101 DATA95,95,95,95,95,95,95,95
32103 DATA250,250,250,250,250,250,250,250
```

```
32107 DATA16,24,252,254,254,252,24,16
32113 DATA0,24,60,126,255,60,60,60
32114 DATA60,60,60,255,126,60,24,0
32115 DATA8,24,63,127,127,63,24,8
32122 DATA250,250,250,250,250,6,252,0
32127 DATA60,66,153,161,161,153,66,60
32176 DATA240,248,252,254,255,255,255,249
32177 DATA7,15,31,63,127,255,255,207
32178 DATA159,159,159,159,158,158,158,159
32179 DATA252,254,255,255,31,15,31,255
32180 DATA240,240,240,240,248,248,252,252,254
32181 DATA135,7,7,15,15,31,31,63
32182 DATA159,159,159,159,158,158,159,159
32183 DATA255,254,252,30,15,15,31,31
32200 DATA1
35000 REM*****************
35010 REM*BILDAUFBAU      *
35020 REM*****************
35030 LET0,0,"■┌────────────┐"
35040 LET0,22,"└────────────┘"
35050 FORI=1TO21:LET0,I,"|*************|":NEXT
35060 FORI=0TO15STEP4
35070 LET15,I,"■┌───────────────────┐"
35080 LET15,I+1,"|                   |"
35090 LET15,I+2,"└───────────────────┘"
35100 NEXTI
35110 LET0,24,"▧CONCENTRATION  ▚ 1984 BY MR-SOFT";
35120 LET15,16,"■┌───────────────────┐"
35130 FORI=1TO5
35140 LET15,16+I,"|                   |":NEXT
35150 LET15,22,"└──────────▐█▌────┘"
35160 LET30,17,"▒▒█▒▒█▒█▒▒▒█H  H▒▒▒▒█▒▒▒Z"
35170 LET17,17,"▩MOEGLICHE"
35180 LET17,18,"RICHTUNGEN"
35190 LET17,20,"H = HILFE"
35200 LET17,21,"* = ENDE "
35210 LET35,23,"▩▩0123▩▩▩▩4567▩"
35220 RETURN
36000 REM*****************
36010 REM*$-AUSGABE       *
36020 REM*****************
36030 FORI=1TO7:FORJ=1TO4
36040 LETJ*3-2,I*3-2,C$(J+(I-1)*4);
36050 NEXTJ,I
36100 RETURN
40000 REM*****************
40010 REM* STRINGS        *
40020 REM*****************
40030 A$="┌──▩▩▩▩ ":B$=" ▩▩▩▩__┘"
40040 DIMB$(28),C$(28),K(28):FORI=1TO26
40050 B$(I)=A$+"▩"+CHR$(64+I)+"▩"+B$
40060 F=1-F:IFF=1THENB$(I)="▩"+B$(I):GOTO40080
40070 B$(I)="▩"+B$(I)
40080 NEXT
40090 B$(28)="▩ ┴ ▩▩▩▩H  H▩▩▩▩ ┬ "
40100 B$(27)="▩"+A$+"▩*▩"+B$
40110 RETURN
41000 REM*****************
41010 REM*MISCHEN         *
41015 REM*****************
41020 FORI=1TO28:K(I)=0:NEXT
41030 FORI=1TO28
41040 X=RND(1)*28+1
41050 IFK(X)=1THEN41040
41060 IFINT(X)=28THENPO=I
41070 K(X)=1:C$(I)=B$(X):NEXT
41080 RETURN
42000 REM*****************
42010 REM*KOMMENTAR       *
42020 REM*****************
42030 POKE646,0:FORI=1TOLEN(K$)
42040 LET16+I,13,MID$(K$,I,1)
42050 FORJ=1TO50:NEXTJ,I:RETURN
43000 REM*****************
43010 REM*RICHTIG         *
43020 REM*****************
43030 FORI=1TO7:FORJ=1TO4
43040 LETJ*3-2,I*3-2,B$(J+(I-1)*4);
43050 NEXTJ,I
43060 K$="RICHTIGE LOESUNG":GOSUB42000
43070 FORI=1TO2000:NEXTI:GOSUB36000
43100 RETURN
```

Maskengenerator

für Apple II

Ein Programm, welches Programme entwickelt: Für unsere fortgeschrittenen Computer-Freunde

Der Begriff Maske wird in der Regel als Kurzform für „Bildschirmmaske" verwendet. Eine Bildschirmmaske ist ein auf dem Bildschirm dargestelltes Formular, in das Daten eingetragen werden können. Gegenüber der üblichen Dateneingabe mittels INPUT-Befehl bietet die Bildschirmmaske eine viel bessere Übersichtlichkeit und oft auch bessere Korrekturmöglichkeiten.
Ein kleines Beispiel: Wird mittels INPUT-Befehl nach dem Namen gefragt, so ist nicht zu erkennen ob der ganze Name oder nur der Nachname gemeint ist. Dies wird erst deutlich, wenn die nächste Frage erscheint. Eine Bildschirmmaske schließt solche Fehler aus, da alle Fragen zu übersehen sind.
In der Regel wird eine Bildschirmmaske mittels PRINT- und GET-Befehl erstellt. Da dies jedoch sehr zeitaufwendig ist, gibt es „Maskengeneratoren". Diese entwickeln die gesamte Bildschirmmaske mit Eingaberoutine.
Bildschirmmasken sind für viele Programme unentbehrlich, da sie die Anzahl der Bedienungsfehler stark reduzieren. Da das Erstellen von Bildschirmmasken jedoch sehr mühselig ist, liegt es nahe, diese vom Computer selbst erstellen zu lassen. Daß dies leicht möglich ist, zeigt unser Programm für den Apple II mit Diskettenlaufwerk.
Nach dem Starten des Programmes wird an einem Beispiel dargestellt, wie einfach die Bildschirmmaske erstellt wird. Danach erscheint folgende Meldung:
ERSTE ZEILEN NR. DER MASKE?
Gemeint ist die Zeilen-Nummer, bei welcher das zu erarbeitende Programm beginnen soll. Die Frage ist dann von Bedeutung, wenn die Maske in ein vorhandenes Programm eingefügt werden soll. Es sollte in diesem Fall eine höhere Zeilen-Nummer gewählt werden als in dem vorhandenen Programm. Nach dem Eingeben der Zeilen-Nummer kommen Sie in den Maskeneditor, der, um eine gute Editiermöglichkeit zu gewährleisten, in Maschinensprache geschrieben ist.
Folgende Befehle sind in dem Editor integriert:

Ctrl A	Cursor 1 Zeile höher
Ctrl Z	Cursor 1 Zeile tiefer
–>	Cursor 1 Stelle weiter
<–	Cursor 1 Stelle zurück
Return	Cursor zum nächsten Zeilenanfang (Rest der Zeile bleibt erhalten)
Ctrl C	Bildschirm wird gelöscht
Ctrl E	Bildschirmmaske erstellen

Diese Befehle sind auch noch einmal im Editor zu sehen. Mit Ctrl C löscht man den Schirm und beginnt die Maske so zu erstellen, wie sie später auszusehen hat. Es ist möglich, den ganzen Schirm bis auf das rechte untere Zeichen auszunutzen, dabei sind alle ASC II verwendbar. Das Zeichen „e" erfüllt eine Sonderfunktion. Überall, wo man dieses antrifft, wird ein Fenster definiert, was später einer Variablen zugeordnet wird. Werden mehrere solcher Zeichen aneinandergereiht, so ist das Fenster dementsprechend breit. Der Maskengenerator erlaubt 100 solcher Fenster mit beliebiger Breite. Die Fenster müssen mindestens durch ein beliebiges Zeichen getrennt werden.
Ist die Maske fertig erstellt, so drücke man Ctrl E und der Maskengenerator beginnt zu arbeiten. Nach etwa 1 Minute möchte der Computer zu jedem Fenster einen Variablennamen zugeordnet haben. Sie können dabei alle Arten von Variablennamen einsetzen. Folgende sind jedoch unzulässig, da diese die Maske selbst benutzt:
LE;TX;TY;ME$;ZE;EI$;GE$.
Selbstverständlich muß man je nach Fenster eine String- oder numerische Variable auswählen.
Wenn alle Fenster einer Variablen zugeordnet sind, wird die Maske erstellt. Die Meldung „MASKE IST ERSTELLT" beendet das Programm.
Soll die Maske in ein Programm eingefügt werden, so muß man erst dieses laden und danach „EXEC MASKE" eingeben.
Die Arbeitsweise der Maske ist relativ einfach. Nach deren Aufruf werden alle Variablen in den entsprechenden Fenstern angezeigt und in dem ersten Fenster wird eine Eingabe erwartet. Tippt man Return, so bleibt der alte Wert bzw. String in der Variablen enthalten und man kommt ins nächste Fenster. Füllt man das Fenster ganz mit Zeichen, so gelangt man ebenfalls ins nächste Feld. Mit der Taste ESC ist es jederzeit möglich, beliebig viele Fenster zurückzuspringen. Fenster, an welchen man sich gerade befindet, werden ganz in Inverse dargestellt, da man so einen guten Überblick über die Fensterbreite besitzt.

```
10  REM  *************************
20  REM  *    MASKENGENERATOR    *
30  REM  *                       *
40  REM  * (C) BEI FRANK BRALL   *
50  REM  *                       *
60  REM  *************************
70  REM
80  ONERR  GOTO 1360
90  HOME
100  PRINT "BILDSCHIRM-MASKENGENERATOR"
110  PRINT : PRINT "         (C) BEI FRANK BRALL"
120  PRINT : PRINT
130  PRINT "DIESE PROGRAMM ERSTELLT EINE ABBRUCH-": PRINT "SICHERE EINGABE
     -MASKE,DIE IN BASIC": PRINT "GESCHRIEBEN IST !"
170  PRINT : PRINT "NACH TASTENDRUCK ERSCHEINT EIN BEISPIEL,";: PRINT "WEL
     CHES MIT DEM EDITOR ERSTELLT WURDE."
180  GET E$: IF E$ = "" THEN 180
```

```
190  HOME : PRINT "KARTEI-NR.: §§§       DATUM: §§§§§§": PRINT "-----------
     ----------------------------": PRINT : PRINT "NAME: §§§§§§§§§§§§  VOR
     NAME §§§§§§§§§§§": PRINT : PRINT "GEBURTSTAG §§§§§§§§ UND ORT §§§§§§§
     §§§§": PRINT : PRINT "LETZTER WOHNORT:§§§§§§§§§§§§§§§§§§§§§§§§§ "
200  PRINT : PRINT : PRINT "HEUTIGER WOHNORT:  PLZ §§§§ §§§§§§§§§§§ ": PRINT
     : PRINT : PRINT "SONSTIGE BEMERKUNGEN: §§§§§§§§§§§§§§§§§§§": PRINT : PRINT
     "§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§§"
210  GET E$: IF E$ = "" THEN 210
220  HOME
310  DIM X(100),Y(100),V$(100),L(100)
320  FOR I = 768 TO 880: READ E: POKE I,E: NEXT I
330  DATA  164,36,169,32,145,40,173,16,192,173,0,192,16,251,72,164,36,177,
     40,73,128,145,40,104,201,133,208,1,96,201,131,208,6,32,88,252,76,0,3,
     201,141,240,10,201,136,240,12,166,36,224,39,208,6
340  DATA  166,37,224,23,240,43,201,149,208,16,166,36,224,39,208,5,169,141
     ,76,79,3,230,36,76,102,3,201,129,208,6,32,26,252,76,102,3,201,154,208
     ,6,32,102,252,76,102,3,32,237,253,164,36,177,40,73,128,145,40,76,6,3
350  DATA   1024,1152,1280,1408,1536,1664,1792,1920,1064,1192,1320,1448,15
     76,1704,1832,1960,1104,1232,1360,1488,1616,1744,1872,2000
360  INPUT "ERSTE ZEILEN-NR. DER MASKE ? ";ZN: HOME
370  PRINT "MASKENEDITOR": PRINT "-------------------------------------- "
380  PRINT : PRINT : PRINT "CTRL A   =  CURSOR 1 ZEILE HOEHER"
390  PRINT : PRINT "CTRL Z   =  CURSOR 1 ZEILE TIEFER"
400  PRINT : PRINT "  ->     =  CURSOR 1 STELLE WEITER"
410  PRINT : PRINT "  <-     =  CURSOR 1 STELLE ZURUECK"
420  PRINT : PRINT "RETURN   =  ZUM NAECHSTEN ZEILENANFANG"
430  PRINT : PRINT "CTRL C   =  SCHIRM LOESCHEN"
440  PRINT : PRINT "CTRL E   =  MASKE ERSTELLEN
450  PRINT : PRINT "§        = MARKIERUNGSZEICHNEN FUER"
460  PRINT : PRINT "           FENSTER;BELIEBIG ANREIHBAR"
470  PRINT : CALL 768: GET E$
480  POKE 36,0: POKE 37,0: PRINT
490 D$ =  CHR$ (4): PRINT D$"DELETE MASKE"
500  CALL 1002
510  PRINT D$"OPEN MASKE"
520  PRINT D$"WRITE MASKE"
530  PRINT ZN"REM **** BILDSCHIRMMASKE ***":ZN = ZN + 1
540  PRINT ZN"GOTO "ZN + 21:ZN = ZN + 1:NR = ZN
550  PRINT ZN"VTAB(TY):HTAB(TX):INVERSE:FOR ZE=1TOLE:?CHR$(32);:NEXTZE":ZN
      = ZN + 1
560  PRINT ZN"VTAB(TY):HTAB(TX):?LEFT$(ME$,LE);":ZN = ZN + 1
570  PRINT ZN"VTAB(TY):HTAB(TX):GE$=" CHR$ (34); CHR$ (34)":ZE=0":ZN = ZN +
     1
580  PRINT ZN"GETEI$:IFEI$<>CHR$(13)ANDEI$<>CHR$(21)THEN"ZN + 3:ZN = ZN +
     1
590  PRINT ZN"IFEI$=CHR$(21)THEN"ZN - 1:ZN = ZN + 1
600  PRINT ZN"GE$=LEFT$(ME$,LE):GOTO"ZN + 13:ZN = ZN + 1
610  PRINT ZN"FORZE=1TOLE:?CHR$(32);:NEXTZE:VTAB(TY):HTAB(TX):ZE=0:GOTO"ZN
      + 2:ZN = ZN + 1
620  PRINT ZN"GET EI$":ZN = ZN + 1
630  PRINT ZN"IF EI$=CHR$(27) THEN GE$=ME$:GOTO"ZN + 10:ZN = ZN + 1
640  PRINT ZN"IF EI$=CHR$(13)THEN"ZN + 8:ZN = ZN + 1
650  PRINT ZN"IF EI$=CHR$(21)THEN"ZN - 3:ZN = ZN + 1
660  PRINT ZN"IF EI$=CHR$(8)ANDZE=0THEN"ZN - 4:ZN = ZN + 1
670  PRINT ZN"IF EI$=CHR$(8)THENZE=ZE-1:GOTO"ZN + 3:ZN = ZN + 1
680  PRINT ZN"IF(LE-1)=ZE THEN GE$=GE$+EI$:ZE=ZE+1:GOTO"ZN + 4:ZN = ZN + 1

690  PRINT ZN"ZE=ZE+1":ZN = ZN + 1: PRINT ZN"? EI$;:GE$=GE$+EI$":ZN = ZN +
     1
700  PRINT ZN"GOTO"ZN - 9:ZN = ZN + 1
710  PRINT ZN"IF GE$<>" CHR$ (34); CHR$ (34)" THEN GE$=LEFT$(GE$,ZE)":ZN =
     ZN + 1
720  PRINT ZN"NORMAL:VTAB(TY):HTAB(TX):FOR ZE=1 TO LE:?CHR$(32);:NEXTZE":Z
     N = ZN + 1
730  PRINT ZN"VTAB(TY):HTAB(TX):?GE$;:RETURN":ZN = ZN + 1
740  PRINT ZN"HOME":ZN = ZN + 1
750  FOR U = 1 TO 24
760  READ A1:ST$ = ""
770  FOR I = A1 TO A1 + 39:C =  PEEK (I) - 128
780  IF C = 64 AND F = 0 THEN F = 1:X(Z) = (I + 1) - A1:Y(Z) = U
790  IF C <  > 64 AND F = 1 THEN F = 0:L(Z) = ((I + 1) - A1) - X(Z):Z = Z +
     1
800  IF C = 64 OR C = 34 THEN C = 32
810 ST$ = ST$ +  CHR$ (C)
820  NEXT I
830  IF F = 1 THEN F = 0:L(Z) = ((I + 1) - A1) - X(Z):Z = Z + 1
```

```
840 I = Ø:T = 4Ø
850  IF  MID$ (ST$,T,1) <   > " " THEN 87Ø
860 I = I + 1:T = T - 1: IF T > Ø THEN 85Ø
870  IF U = 24 AND I < 2 THEN I = 2
880  IF I = 4Ø AND U = 24 THEN  PRINT ZN"PRINT" CHR$ (34); CHR$ (34)";": GOTO
     93Ø
890  IF I = 4Ø THEN  PRINT ZN"PRINT": GOTO 93Ø
900  IF I = Ø THEN  PRINT ZN"PRINT"; CHR$ (34);ST$; CHR$ (34)";": GOTO 93Ø

910  IF U = 24 THEN  PRINT ZN"PRINT"; CHR$ (34);: PRINT  LEFT$ (ST$,41 - I
     );: PRINT  CHR$ (34)";": GOTO 93Ø
920  PRINT ZN"PRINT"; CHR$ (34);: PRINT  LEFT$ (ST$,4Ø - I);: PRINT  CHR$
     (34)
930 ZN = ZN + 1: NEXT U
940  PRINT D$"CLOSE MASKE"
950  IF Z = Ø THEN  HOME : PRINT "FEHLER !!": PRINT : PRINT "ES FEHLT DIE
     KENNZEICHNUNG DER FENSTER !": PRINT : PRINT : PRINT "BEISPIEL:
      DATUM : §§§§§§§ ": PRINT : PRINT "      NAME §§§§§§§§§§§     NR. §§":
      END
960  VTAB 24: HTAB 1: PRINT "                                       ";:U =
     24
970  FOR Q = Ø TO Z - 1
980  IF U = 24 AND Y(Q) = 24 THEN U = 22: VTAB 24: HTAB 1: PRINT  LEFT$ (S
     T$,39);
990  FLASH
1000  VTAB (Y(Q)): HTAB (X(Q)): PRINT "§";
1010  NORMAL
1020  VTAB U: HTAB 1: PRINT "VARIABLENNAME "Q + 1" ?          ";
1030 G$ = "": HTAB 18
1040  GET E$
1050  IF E$ =  CHR$ (8) THEN 1Ø2Ø
1060  IF E$ <   >  CHR$ (13) THEN  PRINT E$;:G$ = G$ + E$: GOTO 1Ø4Ø
1070 V$(Q) = G$: IF G$ =  CHR$ (13) THEN 1Ø2Ø
1080  HTAB (X(Q)): VTAB (Y(Q)): PRINT "§";
1090  NEXT Q
1100  HOME : PRINT D$"MON O"
1110  PRINT D$"APPEND MASKE"
1120  PRINT D$"WRITE MASKE"
1130  FOR Q = Ø TO Z - 1
1140 F = Ø
1150  FOR I = 1 TO  LEN (V$(Q)): IF  MID$ (V$(Q),I,1) = "$" THEN F = 1
1160  NEXT I
1170  IF F = 1 THEN  PRINT ZN"VTAB("Y(Q)"):HTAB("X(Q)"):?LEFT$("V$(Q);","L
     (Q)");"
1180  IF F = Ø THEN  PRINT ZN"VTAB("Y(Q)"):HTAB("X(Q)"):ME$=STR$("V$(Q)"):
     ?RIGHT$( ME$ ,"L(Q)");"
1190 ZN = ZN + 1: NEXT Q:F = Ø
1200  FOR Q = Ø TO Z - 1
1210  PRINT ZN"TX="X(Q)":TY="Y(Q)":LE="L(Q):ZN = ZN + 1
1220  FOR I = 1 TO  LEN (V$(Q)): IF  MID$ (V$(Q),I,1) = "$" THEN F = 1
1230  NEXT I
1240  IF F = 1 THEN  PRINT ZN"ME$="V$(Q)":GOSUB"NR:ZN = ZN + 1
1250  IF F = 1 AND Q > Ø THEN  PRINT ZN"IF EI$ =CHR$(27)THEN"ZN - 6:ZN = Z
     N + 1
1260  IF F = 1 THEN  PRINT ZN;V$(Q)"=GE$":ZN = ZN + 1
1270  IF F = Ø THEN  PRINT ZN"ME$=RIGHT$(STR$("V$(Q)"),"L(Q)"):GOSUB"NR":I
     FGE$=" CHR$ (34); CHR$ (34)" THEN"ZN:ZN = ZN + 1
1280  IF F = Ø AND Q > Ø THEN  PRINT ZN"IF EI$ =CHR$(27)THEN"ZN - 6:ZN = Z
     N + 1
1290  IF F = Ø THEN  PRINT ZN;V$(Q)"=VAL(GE$)":ZN = ZN + 1
1300 F = Ø: NEXT Q
1310  PRINT ZN"REM * ENDE DER MASKE *"
1320  PRINT D$"CLOSE MASKE"
1330  HOME : PRINT "DIE MASKE IST ERSTELLT !": PRINT : PRINT : PRINT "DURC
     H DEN BEFEHLE ";
1340  INVERSE : PRINT "EXEC MASKE";: NORMAL : PRINT : PRINT "WIRD SIE IN D
     EN SPEICHEN GESCHRIEBEN."
1350  NEW : END
1360  REM  *** FEHLERKONTROLLE ***
1370 QE =  PEEK (222)
1380  IF QE = 6 THEN 5ØØ
1390  PRINT  CHR$ (7): PRINT D$"CLOSE MASKE"
1400  PRINT : INVERSE : PRINT "FEHLERCODE "QE: NORMAL : END
```

Labyrinth
für Apple II



Finden Sie den Weg aus dem Labyrinth!

Labyrinth wird auf einem Brett von 15 x 27 Feldern gespielt. Die Ecken der Felder sind durch Punkte gezeichnet. Zwischen diesen Punkten befinden sich „eventuell“ Wände, die Sie am Anfang nicht sehen können. Jedesmal wenn Sie ein neues Feld betreten, werden diese Wände sichtbar. Es stellt also ein großes Labyrinth dar, dessen Glaswände Sie erst nach dem Berühren sehen können. Am Anfang des Spieles werden zwei Kästchen gezeichnet. Das gestreifte Kästchen ist das Ziel, das andere der Start. Sie haben nun die Aufgabe, von dem einen Kästchen zum anderen zu gehen. Dieses geschieht entweder mit Joystick oder Tastature, wobei „I“ = aufwärts, „M“ = abwärts, „J“ = links und „K“ = rechts bedeutet. Sie können auch eine „Schrittbegrenzung“ eingeben, das bedeutet, daß Sie die Anzahl Ihrer Schritte begrenzen können. Die Anzahl der Schritte können Sie dann selber bestimmen.
Das Spiel besitzt zwei Labyrinthe, von denen jeweils eins nach Zufall ausgewählt wird. Auch der Start- und Zielpunkt wird per Zufallszahl bestimmt. Die zwei Labyrinthen sind so geschrieben, daß es nur einen Lösungsweg gibt. Dieser kann aber garantiert werden!

```
10  REM   **********************
20  REM   ** ANDRE JOHANN      **
30  REM   ** EMSSTRASSE 13     **
40  REM   ** 4410 WARENDORF 1 **
50  REM   **********************

60  LOMEM: 16384
70  GOTO 850: REM
80  REM   *** KAESTCHEN PLOTTEN ***
90  HPLOT Y * 10 - 1,X * 10 - 6 TO Y * 10 + 1,X * 10 - 6
100  HPLOT Y * 10 - 1,X * 10 - 5 TO Y * 10 + 1,X * 10 - 5
110  HPLOT Y * 10 - 1,X * 10 - 4 TO Y * 10 + 1,X * 10 - 4
120  RETURN : REM
130  REM  *** STRICHE PLOTTEN ***
140  IF ST(X,Y,1) = 1 THEN  HPLOT Y * 10 - 5,X * 10 - 10 TO Y * 10 + 5,X *
     10 - 10
150  IF ST(X,Y,3) = 1 THEN  HPLOT Y * 10 - 5,X * 10 TO Y * 10 + 5,X * 10
160  IF ST(X,Y,2) = 1 THEN  HPLOT Y * 10 + 5,X * 10 - 10 TO Y * 10 + 5,X *
     10
170  IF ST(X,Y,4) = 1 THEN  HPLOT Y * 10 - 5,X * 10 - 10 TO Y * 10 - 5,X *
     10
180  RETURN : REM
190  REM  *** PRUE.OB ZUG ERL. ***
200 R = 0:S = 0
210  IF V =  - 1 THEN  IF ST(X,Y,1) = 0 THEN R =  - 1
220  IF H = 1 THEN  IF ST(X,Y,2) = 0 THEN S = 1
230  IF V = 1 THEN  IF ST(X,Y,3) = 0 THEN R = 1
```

```
240  IF H =  - 1 THEN  IF ST(X,Y,4) = Ø THEN S =  - 1
250  RETURN : REM
260  REM *** PRUE.OB ZUG ERL. ***
270  IF X < 1 THEN X = 1
280  IF X > 15 THEN X = 15
290  IF Y < 1 THEN Y = 1
300  IF Y > 27 THEN Y = 27
310  RETURN : REM
320 ST(14,22,2) = 1:ST(14,23,4) = 1:ST(8,4,3) = 1:ST(9,4,1) = 1:ST(4,15,2)
     = 1:ST(4,16,4) = 1
330 ST(4,23,3) = Ø:ST(5,23,1) = Ø:ST(1,1Ø,2) = Ø:ST(1,11,4) = Ø:ST(12,16,2
    ) = Ø:ST(12,17,4) = Ø
340  RETURN : REM
350  REM  *** ERST.D.SPIELFE. ***
360  DIM ST(15,27,4)
370  FOR A = 1 TO 15: FOR B = 1 TO 26: READ X:ST(A,B,2) = X:ST(A,(B + 1),4
    ) = X: NEXT B,A
380  FOR A = 1 TO 14: FOR B = 1 TO 27: READ X:ST(A,B,3) = X:ST((A + 1),B,1
    ) = X: NEXT B,A
390  HGR : HCOLOR= 7
400  HPLOT 5,Ø TO 275,Ø TO 275,15Ø TO 5,15Ø TO 5,Ø
410  FOR A = 5 TO 275 STEP 1Ø: FOR B = Ø TO 15Ø STEP 1Ø: HPLOT A,B: NEXT B
    ,A: REM
420 GH =  RND (1): IF GH > .5 THEN  GOSUB 32Ø: REM

430  REM  *** ZIEL KOORDIN. ***
440 X =  INT ( RND (1) * 15) + 1:Y =  INT ( RND (1) * 27) + 1
450  HCOLOR= 1: GOSUB 9Ø: GOSUB 14Ø: HCOLOR= 7:N = X:M = Y: REM
460  REM  *** ANFANG KOORD. ***
470 X =  INT ( RND (1) * 15) + 1:Y =  INT ( RND (1) * 27) + 1
480  GOSUB 9Ø: GOSUB 14Ø:SB = Ø
490  IF JN$ = "J" THEN  VTAB 23: HTAB 1: INPUT "SCHRITTZAHL : ";SZ:SZ =  INT
    (SZ)
500  HOME
510  IF JK$ = "J" THEN 64Ø: REM
520  REM *** STEUERUNG TAST. ***
530  WAIT  - 16384,128: GET R$:H = Ø:V = Ø
540  IF R$ = "J" THEN H =  - 1
550  IF R$ = "K" THEN H = 1
560  IF R$ = "I" THEN V =  - 1
570  IF R$ = "M" THEN V = 1
580  GOSUB 2ØØ: IF R = Ø AND S = Ø THEN 53Ø
590  HCOLOR= Ø: GOSUB 9Ø: HCOLOR= 7
600 X = X + R:Y = Y + S: GOSUB 27Ø: GOSUB 9Ø: GOSUB 14Ø: GOSUB 1Ø6Ø
610  IF X = N AND Y = M THEN 79Ø
620  GOTO 53Ø: REM
630  REM  *** STEUERUNG JOYS. ***
640  FOR I = 1 TO 1ØØ: NEXT
650 A =  PDL (Ø): FOR I = 1 TO 1Ø: NEXT :B =  PDL (1)
660 H = Ø:V = Ø
670  IF A < 5Ø THEN H =  - 1
680  IF A > 23Ø THEN H = 1
690  IF B < 7Ø AND H = Ø THEN V =  - 1
700  IF B > 25Ø AND H = Ø THEN V = 1
710  GOSUB 2ØØ
720  IF R = Ø AND S = Ø THEN 64Ø
730  HCOLOR= Ø: GOSUB 9Ø: HCOLOR= 7
740 X = X + R:Y = Y + S
750  GOSUB 27Ø: GOSUB 9Ø: GOSUB 14Ø: GOSUB 1Ø6Ø
760  IF X = N AND Y = M THEN 79Ø
770  GOTO 64Ø: REM
780  REM ***     ENDE      ***
790  FOR A = 1 TO 5: HCOLOR= Ø: GOSUB 9Ø: HCOLOR= 7: GOSUB 9Ø: PRINT  CHR$
    (7): NEXT
800  TEXT : HOME : VTAB 3: IF SB = SZ THEN  PRINT "DIE ZEIT IST UM ! SIE H
    ABEN VERLOREN": PRINT : PRINT : GOTO 82Ø
810  PRINT "SIE HABEN GEWONNEN !!! - GRATULATION": PRINT : PRINT
820  PRINT "MOECHTEN SIE NOCH EINMAL SPIELEN ? ";: GET P$: IF P$ = "J" THEN
     RUN
830  HOME : PRINT "TSCHUESS...": END
840  REM  ***    ANFANG     ***
850  TEXT : HOME : FOR A = 1 TO 4Ø:A$ = A$ + "*": NEXT
860 B$ = "*": FOR A = 1 TO 38:B$ = B$ + " ": NEXT :B$ = B$ + "*"
870  INVERSE : PRINT A$;: FOR A = 1 TO 1Ø: PRINT B$;: NEXT : PRINT A$
```

```
880  FLASH : VTAB 3: HTAB 16: PRINT "LABYRINTH": INVERSE
890  VTAB 6: HTAB 9: PRINT "GES. V.    ANDRE JOHANN"
900  VTAB 8: HTAB 7: PRINT "EMSSTR.13    4410 WARENDORF"
910  VTAB 10: HTAB 9: PRINT "(C) BY     OKTOBER 1984"
920  NORMAL
930  VTAB 19: PRINT "REGELN ? (J/N) ";: GET RR$
940  IF RR$ = "J" THEN  GOSUB 1000
950  VTAB 21: HTAB 1: INVERSE : PRINT "J";: NORMAL : PRINT "OYSTICK ODER "
     ;: INVERSE : PRINT "K";: NORMAL : PRINT "EYBOARD : ";
960  GET JK$
970  VTAB 23: HTAB 1: PRINT "SCHRITTBEGRENZUNG ? (J/N) :";: GET JN$
980  HOME : VTAB 12: HTAB 14: PRINT "BITTE WARTEN"
990  GOTO 360
1000  HOME : INVERSE : PRINT "* * * * * * * *  REGELN  * * * * * * * *": NORMAL

1010  PRINT "IN DIESEM SPIEL HABEN SIE DIE AUFGABE": PRINT : PRINT "DURCH
     EIN GLASLABYRINTH ZU DEM GESTREIF-": PRINT "TEN KAESTCHEN ZU GEHEN.SI
     E KOENNEN ABER": PRINT : PRINT "DIE WAENDE NICHT SEHEN,ERST WENN SIE
     SIE"
1020  PRINT "BERUEHREN WISSEN SIE WO SIE SIND.": PRINT : PRINT "SIE HABEN
     DIE MOEGLICHKEIT EINE SCHRITT-": PRINT "BEGRENZUNG EINZUGEBEN,IN DER
     SIE": PRINT : PRINT "VERSUCHEN,DIE AUFGABE ZU LOESEN"
1030  PRINT : PRINT " =AUFWAERTS  =ABWAERTS  =LINKS  =RECHTS";
1040  INVERSE : HTAB 1: PRINT "I";: HTAB 13: PRINT "M";: HTAB 24: PRINT "J
     ";: HTAB 32: PRINT "K": NORMAL
1050  RETURN
1060 SB = SB + 1: IF JN$ = "J" THEN  VTAB 23: HTAB 1: PRINT "SCHRITTE : ";
     SB: IF SB = SZ THEN 790
1070  RETURN
1080  DATA 0,0,0,1,0,1,0,0,1,1,1,0,0,0,1,0,0,0,1,0,0,1,0,0,1,0,1,0,1,1,1,0
     ,0,0,0,1,1,0,0,0,1,1,0,1,0,1,0,0,0,1,1,1,1,1,0,0,0,0,0,0,1,0,0,1,0,0,
     1,0,0,0,1,1,1,0,0,0,1,1
1090  DATA 1,0,0,0,0,1,1,0,0,0,0,1,0,0,0,1,1,0,1,0,0,1,0,0,0,0,0,1,0,0,0,1
     ,1,1,0,0,1,1,1,1,0,0,1,0,0,1,1,1,1,1,0,0,1,0,0,1,1,1,0,1,0,1,1,0,1,1,
     0,0,0,0,1,1,0,1,0,0,0,0
1100  DATA 0,1,0,1,0,1,0,0,0,1,0,0,0,0,0,1,0,0,1,0,1,0,0,0,1,1,0,0,1,1,1,0
     ,0,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,1,0,0,1,0,1,1,1,1,0,0,1,0,1,1,1,
     0,0,0,0,1,0,1,1,1,0,0,1
1110  DATA 1,0,0,0,1,0,0,0,1,1,0,0,0,1,1,0,1,1,0,1,0,1,0,0,1,1,1,0,0,1,1,0
     ,1,1,1,1,0,0,0,0,1,0,0,1,0,0,0,1,0,1,0,0,1,1,0,1,1,1,1,1,0,0,0,1,0,1,
     0,1,1,1,1,0,1,0,1,1,0,1
1120  DATA 1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,1,0,0,1,1,0,0,1,1,0,1,0
     ,0,0,1,0,0,0,0,0,0,1,1,0,1,0,0,0,1,0,1,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,
     0,0,0,1,0,0,0,0,0,1,0,0
1130  DATA 0,0,1,1,0,0,0,1,1,0,0,0,1,1,1,0,1,1,1,0,1,1,1,0,1,0,0,0,0,0,0,0
     ,1,1,1,1,1,0,1,1,1,0,0,0,1,1,0,0,1,1,1,0,1,0,1,0,1,1,1,1,0,1,1,0,1,1,
     0,1,1,0,1,0,1,0,0,0,1,0,1,0,0
1140  DATA 0,0,1,1,1,1,0,1,0,1,1,0,0,1,0,1,0,0,1,0,0,1,1,1,1,1,1,0,1,0,1,1
     ,0,0,0,0,1,0,0,0,0,1,1,1,1,1,0,0,0,0,0,1,1,0,0,1,0,1,0,0,1,0,1,1,0,1,
     1,0,1,1,1,1,0,1,0,1,0,1,1,1,1
1150  DATA 1,0,1,0,1,0,1,0,0,1,0,1,1,1,1,0,0,0,0,1,0,1,0,1,1,0,0,1,1,1,0,0
     ,0,1,1,1,0,0,0,1,0,0,1,0,1,1,0,0,0,0,1,0,1,0,0,1,1,0,0,1,0,1,1,0,0,1,
     0,0,1,1,0,1,0,1,0,0,0,0,1,0,0
1160  DATA 0,0,1,1,1,0,1,0,0,0,0,1,1,1,0,0,1,0,0,1,0,1,1,0,1,0,0,0,0,1,1,0
     ,0,0,1,0,0,1,1,1,1,0,1,0,1,0,1,1,1,1,0,1,1,0,0,0,0,0,0,0,1,0,0,1,0,0,
     0,0,0,1,0,0,1,0,1,0,1,0,1,0,0
1170  DATA 0,0,0,0,1,1,0,0,1,0,1,0,1,0,1,1,0,1,1,1,0,1,0,1,0,0,1,0,1,0,1,1
     ,0,0,1,1,1,0,1,1,1,1,0,1,0,0,1,1,1,1,0,0,1,0
```

DATA-Generator

für Apple II

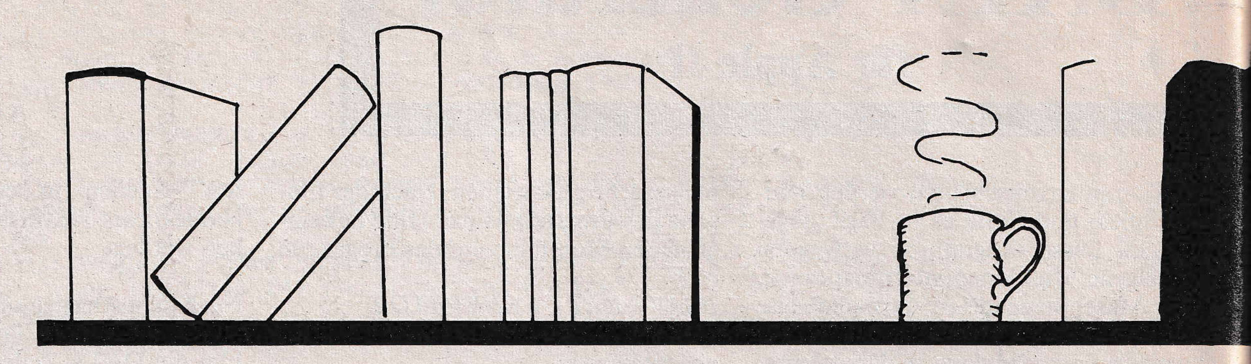
In Computerzeitschriften werden häufig Spiele und Utilities in Basic abgedruckt. Diese Programme verwenden oft eigene Maschinenunterprogramme. Der Maschinencode ist normalerweise in DATA-Zeilen im Basic-Programm abgelegt und wird dann in den entsprechenden Speicher gepoked.
Der DATA-Generator für denApple II (+/e/c) und kompatible Computer verarbeitet Maschinencode zu solchen DATA-Zeilen. Damit hat jeder nun die Möglichkeit, Maschinenprogramme und Basic-Programme zusammenzufügen.
Der DATA-Generator ist in Maschinensprache geschrieben. Nach der Eingabe des Programms mit Hilfe des eingebauten Monitors wird das Programm mit:
BSAVE DATA-GENERATOR,
AiEB
abgespeichert.
Gestartet wird es dann mit CALL 32768 oder BRUN DATA-GENERATOR.
Anschließend erfolgt die Eingabe des Adreßbereichs, der in DATA-Zeilen abgelegt werden soll. Die Eingabe muß im Hexadezimal-Format erfolgen und genau vier Stellen haben.
Der DATA-Generator schreibt die DATA-Zeilen in ein Textfile. Der Name dieses Textfiles muß nun angegeben werden.

Nach dem Ablegen sämtlicher Bytes im Datenfile wird das Programm beendet. Um die DATA-Zeilen nun einzulesen, gibt man „EXEC Name" (Name des Datenfiles) ein.

Startet man den DATA-Generator mit CALL 33768, beginnen die Zeilennummern oberhalb der zuletzt genutzten. Das bietet die Möglichkeit, mehrere Maschinenunterprogramme als DATA-Zeilen aneinanderzufügen.

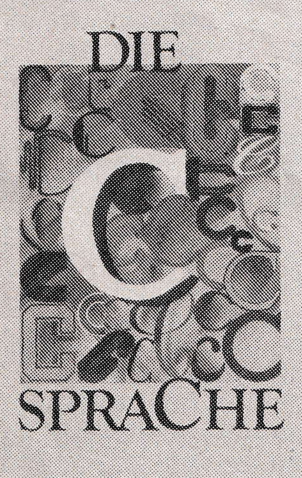
```
*8ØØØ.81EA

8ØØØ- A9 AØ 85 33 AD B6 AA 85
8ØØ8- E7 A9 ØØ 8D B6 AA 2Ø 58
8Ø1Ø- FC A9 37 AØ 81 2Ø 3A DB
8Ø18- A9 Ø5 85 25 2Ø 22 FC A9
8Ø2Ø- 88 AØ 81 2Ø 3A DB 2Ø 6F
8Ø28- FD EØ Ø4 DØ EB AØ ØØ 2Ø
8Ø3Ø- A7 FF A5 3E 85 E3 A5 3F
8Ø38- 85 E4 2Ø 8E FD 2Ø 8E FD
8Ø4Ø- A9 A2 AØ 81 2Ø 3A DB 2Ø
8Ø48- 6F FD EØ Ø4 DØ EC AØ ØØ
8Ø5Ø- 2Ø A7 FF A5 3E 85 E5 A5
8Ø58- 3F 85 E6 E6 E5 DØ Ø2 E6
8Ø6Ø- E6 2Ø 8E FD 2Ø 8E FD A9
8Ø68- BC AØ 81 2Ø 3A DB 2Ø 6F
8Ø7Ø- FD E8 8A A8 BD FF Ø1 9D
8Ø78- 5Ø Ø2 CA DØ F7 C8 A9 ØØ
8Ø8Ø- 99 5Ø Ø2 A9 C9 AØ 81 2Ø
8Ø88- 3A DB A9 51 AØ Ø2 2Ø 3A
8Ø9Ø- DB A9 DØ AØ 81 2Ø 3A DB
8Ø98- A9 51 AØ Ø2 2Ø 3A DB AØ
8ØAØ- ØØ 84 EB A5 EB DØ Ø3 2Ø
8ØA8- 14 81 C6 EB AØ ØØ B1 E3
8ØBØ- A8 A9 ØØ 85 E8 85 E9 85
8ØB8- EA CØ ØØ FØ 16 A2 Ø3 CA
8ØCØ- F6 E8 B5 E8 C9 ØA 9Ø Ø8
8ØC8- A9 ØØ 95 E8 EØ ØØ DØ EF
8ØDØ- 88 DØ EA A5 E8 FØ Ø5 Ø9
8ØD8- BØ 2Ø ED FD A5 E9 Ø9 BØ
8ØEØ- 2Ø ED FD A5 EA Ø9 BØ 2Ø
8ØE8- ED FD A5 EB FØ Ø5 A9 AC
8ØFØ- 2Ø ED FD E6 E3 DØ Ø2 E6
8ØF8- E4 A5 E4 C5 E6 9Ø A4 A5
81ØØ- E3 C5 E5 DØ 9E A9 D8 AØ
81Ø8- 81 2Ø 3A DB A5 E7 8D B6
811Ø- AA 4C Ø3 EØ A9 32 85 EB
8118- 2Ø 8E FD A2 Ø4 FE E1 81
812Ø- BD E1 81 C9 3A DØ Ø8 A9
8128- 3Ø 9D E1 81 CA DØ EE A9
813Ø- E1 AØ 81 2Ø 3A DB 6Ø C4
8138- C1 D4 C1 AD C7 C5 CE C5
814Ø- D2 C1 D4 CF D2 AØ AØ A8
8148- C3 A9 AØ B1 B9 B8 B4 AØ
815Ø- AØ CF AE AØ D3 D4 C5 C9
8158- CE CD C5 C9 C5 D2 AØ AD
816Ø- AD AD AD AD AD AD AD AD
8168- AD AD AD AD AD AD AD AD
817Ø- AD AD AD AD AD AD AD AD
8178- AD AD AD AD AD AD AD AD
818Ø- AD AD AD AD AD AD AD ØØ
8188- D3 D4 C1 D2 D4 C1 C4 D2
819Ø- C5 D3 D3 C5 AØ A8 C9 CE
8198- AØ C8 C5 D8 A9 AØ BA AØ
81AØ- A4 ØØ C5 CE C4 C1 C4 D2
81A8- C5 D3 D3 C5 AØ A8 C9 CE
81BØ- AØ C8 C5 D8 A9 BA AØ AØ
81B8- AØ AØ A4 ØØ C6 C9 CC C5
81CØ- AD CE C1 CD C5 AØ BA AØ
81C8- ØØ 8D 84 CF DØ C5 CE ØØ
81DØ- 8D 84 D7 D2 C9 D4 C5 ØØ
81D8- 8D 84 C3 CC CF D3 C5 8D
81EØ- ØØ 36 3Ø 3Ø 3Ø 3Ø 44 41
81E8- 54 41 ØØ
*
```



Homecomputer-Bücherkiste

Stanke/Lösch
Die C-Sprache



Mit diesem Buch von Zdenek Stanke und Stefan Lösch wird dem Leser der Einstieg in die Programmiersprache C leichtgemacht.
C ist eine Sprache der Zukunft, die sogar Pascal bereits ernsthafte Konkurrenz macht. Sie verbindet einige Vorteile der herkömmlichen Programmiersprachen mit neuen Konzepten und Vorzügen.
Die Autoren, beide erfahrene Praktiker, haben es sich zur Aufgabe gemacht, sowohl dem Programmierneuling als auch den Experten schrittweise an die Benutzung und die Möglichkeiten dieser Allzweck-Programmiersprache heranzuführen.

Der Leser findet im Text eingefügt zahlreiche Beispielprogramme und Flußdiagramme, mit deren Hilfe es den Autoren gelingt, auch komplexe Zusammenhänge gut verständlich zu machen.
Neben einer fundierten Einführung in die C-Sprache bietet gerade das Kapitel über die verschiedenen Bibliotheksfunktionen auch für den Experten eine äußerst nützliche Informationsquelle.
Das Buch eignet sich von seiner Konzeption her sowohl als Lehrbuch als auch als Nachschlagewerk.

Die Daten:
TeWi-Verlag, München
288 Seiten mit vielen Programmierbeispielen,Softcover, Preis DM 59,-
ISBN 3-921 803-28-4

Radeloff
Software-System Handbuch für Laser-Heimcomputer

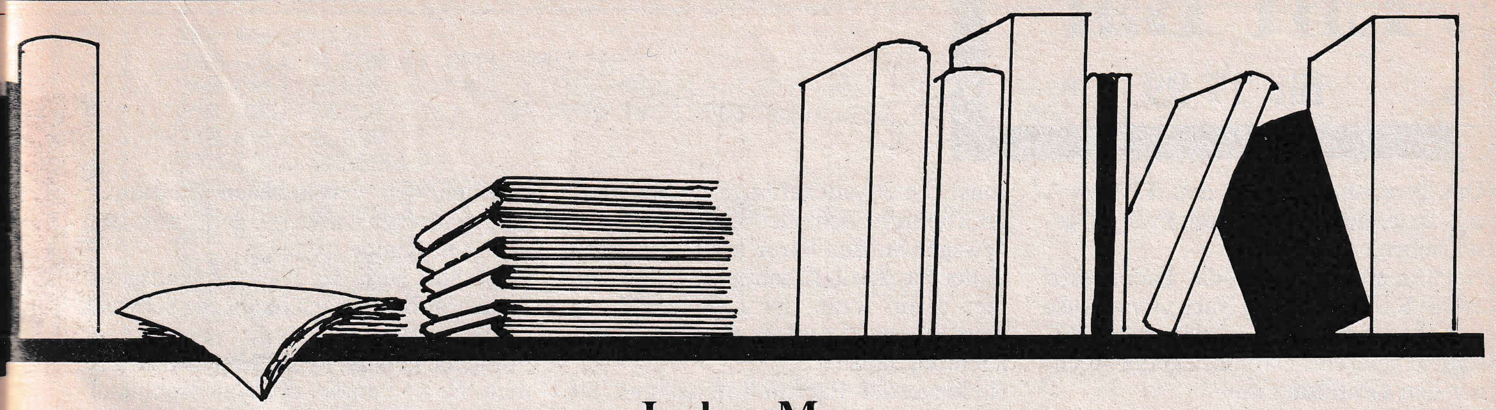


Das Buch ist die ideale Ergänzung zum Basic-Handbuch der Laser-Computer.
Fragen und Probleme, die bisher noch nicht besprochen wurden, sind in dem Buch ausreichend und verständlich erklärt.

Viele Programmiertips, z.B. Sparen von Speicherplatz, erleichtern dem Anwender das strukturierte Programmieren.

Neben Erläuterungen zum Maschinencode sind auch die gut dokumentierten Schaltpläne lobenswert zu erwähnen.
Im übrigen läßt sich das Buch auf die verschiedenen Laser-Versionen 110/ 210/310 und VZ-200 anwenden und wird somit für den Laser-Besitzer eine sehr interessante Hilfestellung. Das Software-System Handbuch ist im Fachhandel, teilweise auch im Buchhandel erhältlich.

Die Daten:
Sanyo-Video Vertrieb (Hamburg), 1984, 98 Seiten, DM 35,-



Jochen Merz

Maschinencode-Handbuch für den ZX-Spectrum

Für viele Spectrum-Anwender ist die Maschinensprache noch immer ein Buch mit sieben Siegeln.
Dieses Buch verschafft einen Einblick in die Welt des Z-80 Mikroprozessors und seiner Programmierung.

Dabei wurde auf überflüssige technische Erläuterungen verzichtet, dafür in seinem Inhalt und Aufbau leicht verständlicher Text gebracht.
Der Bezug zum Basic wurde sehr gut erläutert, sodaß die doch sehr schwierige Assemblersprache etwas durchsichtiger wird.
Das Buch enthält neben dem kompletten Z-80 Befehlssatz auch eine Dez/Hex Umrechnungstabelle.
Dem Maschinensprache-Einsteiger wird dieses Buch viel Freude bereiten.

Die Daten:
Profisoft 1983, 76 Seiten,
ISBN 3-923985-02-9

Sharp

Pocket-Computer PC1500/1500A

System-Handbuch Maschinensprache

Als Sharp im Sommer 1981 seinen Pocket-Computer PC 1500 der Weltöffentlichkeit präsentierte, kam mit diesem Taschengerät eine Maschine auf den Markt, die wesentlich mehr leistete, als seinerzeit einem Pocketcomputer zugetraut wurde. Bereits das komfortable 24K-Basic ließ ahnen, daß dieser Rechner angetreten war, es mit weitaus größeren Tischgeräten aufzunehmen. Kein Wunder, daß viele Anwender sehnsüchtig auf eine Einführung in die Maschinensprache des PC1500 (bzw. des Nachfolge-Modelles 1500A) warteten.

Seit geraumer Zeit ist eine Sharp-Dokumentation auf dem Markt, die im Verlag Günter Holtkötter GmbH Hamburg erschien und sehr ausführlich in die Welt des Prozessors LH 5801 einführt.
Der LH 5801 ist ein statischer 8-Biter, der die Vorzüge der CMOS-Technologie mit einer beachtlichen Datendurchsatz-Kapazität verbindet. 128K sind direkt adressierbar. Der LH 5801 verwendet ein 6-Byte-Universalregister, das mit dem Akkumulator drei Paare von 2-Byte-Adresspointern zu benutzen erlaubt.

9-Bit Timer, drei Interrupt-Arten (maskiert, unmaskiert, timer-orientiert), WAIT-Funktion, Clock-PO für Inputport und external latch, Memory-Backup (BFI/BFO), LCD-Backplate, Programm-Optimierung durch 28 verschiedene einschrittige Vektorsprünge sind Merkmale einer hochentwickelten Prozessor-Technologie, deren Verständnis umfangreiches Spezialwissen erforderlich macht.

Das vorliegende Handbuch geht didaktisch einwandfrei und sehr präzise an den breitgefächerten Stoff heran und liefert alles, was der angehende LH 5801-Systemprogrammierer benötigt.

Die Daten:
Sharp/Holtkötter 1982, 171 S., 2 Faltkarten, DM 69,- (Vertrieb über Fachhandel).

Thorsten Ritter

Erfolg mit Multiplan

Dieses Buch ist eine sehr große Hilfe für Leute, die mit Multiplan arbeiten wollen.

Schritt für Schritt werden Ihnen anhand zahlreicher Beispiele die einzelnen Elemente von Multiplan erläutert. Sie werden nach kurzer Zeit in der Lage sein, Multiplan optimal für geschäftliche, wissenschaftliche und allgemeine Zwecke einzusetzen.
Ein wirklich empfehlenswertes Buch.

J. W. Coffran

Programmierung des 8086/8088

Dieses Buch gibt eine umfangreiche Einleitung über die Programmierung der meisten PC-Computer in Assembler.

Es ist sowohl für Anfänger als auch für fortgeschrittene Programmierer geeignet. Eingegangen wird auf Grundlagen der Programmierung, den internen Aufbau des 8086/8088, die Speicherorganisation, die Adressierungsarten und den Befehlssatz mit individueller Beschreibung. Außerdem werden grundlegende Programmiertechniken z.B. Unterberechnungen und Ein- und Ausgabetechniken ausführlich erläutert. Dazu kommen noch Anwendungsbeispiele für den IBM-Personalcomputer und Hilfen für die Programmentwicklung. Wir können dieses Buch nur empfehlen.

Die Daten:
ISBN-3-88745-050-7

Lift Bär

für TI-99

Das Programm ist für den TI 99/4 A mit Extended BASIC und Joystick geschrieben.
Nach dem Eintippen oder Laden wird es mit RUN gestartet. Nach einer Pause von ca. 3 Sekunden wird der Bildschirm gelöscht. Dann belegt der Rechner den Grafiksatz um.
Der Bildschirm wird jetzt schwarz, und das Titelbild erscheint in Multicolor-Großbuchstaben. Dabei wird Song of Joy gespielt.
Sie können mit der Taste „J" eine kurze Spielanleitung abrufen. Bei jeder anderen Taste erhalten Sie keine Spielanleitung.
In der Stadt wird ein Neubau errichtet. Es ist kurz nach Feierabend, und die Maurer haben ihre Arbeit beendet. Sie haben jedoch einen Teil ihres Werkzeuges vergessen und einen kleinen Bären beauftragt das Werkzeug für sie zu holen. Bei dem Werkzeug handelt es sich um 2 Hämmer, 2 Gabelschlüssel und 2 Schraubendreher.
Die Aufgabe hört sich zwar leicht an, ist aber gar nicht so einfach zu lösen, da erstens der Bär nicht weiß wo das Werkzeug versteckt ist, zweitens die einzelnen Etagen des Neubaues nur mittels auf- und abfahrenden Fahrstühlen zu erreichen sind und drittens nur eine befristete Zeit zur Verfügung steht.
Sie können den Bären mit dem Joystick nach rechts und links bewegen. Achten Sie jedoch darauf, daß die Taste „ALPHA LOCK" nicht gedrückt ist, da sonst die Joystickabfrage nicht richtig funktioniert. Zum Betreten eines Liftes bewegen Sie den Bären in die Richtung, in der Sie den Lift betreten wollen, immer dann, wenn der Lift gerade vor dem Bären ist und in die gewünschte Richtung fährt.
So lange der Bär im Lift ist und Sie nicht aussteigen wollen, muß der Joystick in der neutralen Mittelstellung stehen.
Zum Verlassen eines Liftes drücken Sie den Joystick wieder in die gewünschte Richtung.
Der Bär soll ja, wie schon zuvor erwähnt, Werkzeug suchen. Das Werkzeug ist in den 12 Fragezeichen versteckt. Zum Aufheben eines Fragezeichens führen Sie den Bären genau darüber und drücken dann den Aktionsknopf am Joystick.
Wenn sich unter dem Fragezeichen ein Werkzeug befindet, so ertönt ein Multiklang-Gong, und das Werkzeug erscheint in der oberen linken Bildschirmecke. Befindet sich unter dem aufgenommenen Fragezeichen jedoch kein Werkzeug, so werden Sie ein Kratzen vernehmen und es kann sein, daß die Wände des Neubaues versetzt werden. Dies kann Ihnen einen günstigeren Weg verschaffen, es kann Ihnen jedoch auch den Weg versperren, so daß Sie einen Umweg zum nächsten Fragezeichen wählen müssen.
Wenn Sie alle 6 Werkzeuge gefunden haben, so bekommen Sie zu Ihren Punkten den verbliebenen Zeitbonus gutgeschrieben, und es geht mit der nächsten Runde weiter.
Insgesamt wird der Schwierigkeitsgrad 7mal erhöht. Danach bleibt die Schwierigkeit konstant.
Für jeden gefundenen Hammer bekommen Sie 80 Punkte, der Gabelschlüssel bringt 100 Punkte und der Schraubendreher sogar 120 Punkte. Für alle 5000 erreichten Punkte erhalten Sie ein Bären-Bonusleben gutgeschrieben. Der Bär verliert sein Leben wenn er
1. zu hoch oder zu tief aus dem Lift springt,
2. gegen eine vertikale Mauer läuft,
3. mit dem Lift versucht durch eine Mauer zu fahren,
4. den Bildschirm nach irgend einer Seite verläßt,
5. wenn der Zeitbonus auf 0 abgelaufen ist.

Die Fahrstühle fahren immer auf und ab. Bei jedem Richtungswechsel wird der anfänglich 3500 Punkte zählende Zeitbonus um 50 Punkte verringert. Mit steigendem Schwierigkeitsgrad fahren die Lifte immer schneller auf und ab.
Somit läuft auch der Zeitbonus immer schneller, und das Ein- und Aussteigen wird immer schwieriger. Nachfolgend ist das Programm in groben Zügen beschrieben. Den exakten und strukturierten Programmaufbau können erfahrene Programmierer der beigefügten Zuordnungstafel entnehmen.

```
100 !@P-
110 REM ******************************
120 REM *    L I F T  -  B A E R     *
130 REM *----------------------------*
140 REM * Ein Spiel fuer TI 99/4 A   *
150 REM * in TI Extended BASIC       *
160 REM * mit Joystick zu steuern    *
170 REM *----------------------------*
180 REM * (c) 1984 by Volker Becker  *
190 REM *      Steinbacher Str. 10   *
200 REM *      6370 Oberursel 6      *
210 REM ******************************
220 REM
230 REM
240 !@P+
250 OPTION BASE 1
260 DIM VFELD(6),XFELD(9),SF(4),WFELD(3)
270 GOTO 310 :: A :: B :: BER :: BONUS :: BV :: C :: F :: I :: K :: MOT :: P ::
RUNDE :: S :: ST :: SVAR :: T :: V :: VWERT :: X :: X1 :: X2 :: Y :: Z
280 CALL ASCIISET :: CALL BAER :: CALL CLEAR :: CALL COINC :: CALL COLOR :: CALL
 DELSPRITE :: CALL EINLEITUNG :: CALL GCHAR :: CALL GONG :: CALL HCHAR
290 CALL JOYST :: CALL KEY :: CALL LOCATE :: CALL MAGNIFY :: CALL MELODY :: CALL
 MOTION :: CALL PATTERN :: CALL POSITION :: CALL PUNKTE :: CALL SCREEN
300 CALL SONGOFJOY :: CALL SOUND :: CALL SPRITE :: CALL TITEL :: CALL VCHAR :: C
ALL ZEIT :: !@P-
310 CALL CLEAR :: CALL COLOR(5,1,16,6,1,16,7,1,16,8,1,16):: CALL COLOR(3,16,1,4,
16,1,2,14,1,9,6,16)
```

```
320 RANDOMIZE
330 CALL CLEAR :: CALL ASCIISET
340 CALL CLEAR :: CALL TITEL(XFELD(),5,SF(),0,2):: CALL CLEAR
350 CALL KEY(5,T,S):: IF T=74 OR T=106 THEN CALL EINLEITUNG
360 CALL CLEAR
370 CALL SCREEN(2)
380 FOR I=4 TO 20 STEP 4
390 CALL HCHAR(I,4,112,12):: CALL HCHAR(I,18,112,12)
400 NEXT I
410 CALL HCHAR(24,1,113,32)
420 CALL VCHAR(1,1,112,24):: CALL VCHAR(1,32,112,24)
430 !@P+
440 DATA 1,8,120,1,12,128,5,6,128,5,11,136,5,23,120,5,27,136,9,4,120,9,29,128
450 !@P-
460 DATA 13,9,128,13,13,136,13,18,120,13,24,136,17,12,120,17,22,128,17,27,136,21
,4,136,21,15,128,21,28,120
470 DATA 8,2,120,8,16,120,8,30,136,12,2,128,12,16,136,12,30,128,16,2,136,16,16,1
28,20,30,120
480 RESTORE 440
490 FOR I=1 TO 18 :: READ A,B,C :: CALL VCHAR(A,B,C,3):: NEXT I
500 FOR I=1 TO 9 :: READ A,B,C :: CALL HCHAR(A,B,C,2):: NEXT I
510 CALL COLOR(11,7,2,13,7,2,12,1,1,14,1,1,9,6,16):: CALL SCREEN(5)
520 RUNDE=1 :: BER=4 :: Z=3500 :: P=0 :: BONUS=1 :: CALL PUNKTE(P):: CALL ZEIT(Z
):: CALL BAER(BER)
530 CALL MAGNIFY(3)
540 CALL SPRITE(#2,116,13,177,9,#3,116,13,1,121,#4,116,13,177,233)
550 MOT=RUNDE+4 :: CALL MOTION(#2,-MOT,0,#3,MOT,0,#4,-MOT,0)
560 DATA 3,10,3,26,7,9,7,25,11,13,11,23,15,11,15,21,19,8,19,25,23,13,23,22
570 GOSUB 820
580 BV=128
590 CALL SPRITE(#1,124,11,169,41)
600 ON ERROR 730
610 CALL ZEIT(Z)
620 REM HauPtschleife
630 V=V+1 :: CALL POSITION(#3,X,Y):: IF X>200 THEN GOSUB 940
640 CALL JOYST(1,X,Y):: IF X=0 THEN 660 ELSE IF X=-4 THEN CALL PATTERN(#1,124)EL
SE CALL PATTERN(#1,132)
650 CALL SOUND(-5,523,0)
660 CALL MOTION(#1,0,2*X)
670 CALL COINC(ALL,ST):: IF ST THEN 970
680 CALL POSITION(#1,X,Y):: X=(X+7)/8+2 :: Y=(Y)/8+1 :: CALL GCHAR(X,Y,X1)
690 IF X1=BV OR X1=112 OR X1=113 THEN 700 ELSE 730
700 CALL GCHAR(X-2,Y,X1):: CALL GCHAR(X-2,Y+1,X2):: IF X1=BV OR X2=BV THEN 730
710 CALL KEY(1,T,S):: IF T=18 THEN 1080
720 GOTO 620
730 FOR I=1 TO 4 :: CALL MOTION(#I,0,0):: NEXT I
740 CALL PATTERN(#1,140)
750 CALL MELODY
760 BER=BER-1 :: CALL BAER(BER)
770 IF BER<0 THEN CALL DELSPRITE(ALL):: BER=0 :: CALL BAER(BER):: CALL TITEL(XFE
LD(),1,SF(),-1,5):: GOTO 370
780 IF SVAR THEN Z=3500 :: SVAR=0 :: GOTO 1370
790 CALL MOTION(#2,-MOT,0,#3,MOT,0,#4,-MOT,0)
800 SVAR=0
810 GOTO 590
820 RESTORE 560
830 FOR I=1 TO 6
840 VFELD(I)=2
850 NEXT I
860 FOR I=1 TO 12
870 READ A,B
880 F=INT(RND*5+1.4):: IF VFELD(F)=0 THEN 880
890 VFELD(F)=VFELD(F)-1
900 CALL HCHAR(A,B,F+95)
910 NEXT I
920 FOR K=1 TO 3 :: WFELD(K)=0 :: NEXT K
930 RETURN
940 IF V<5 THEN RETURN ELSE V=0 :: MOT=-MOT :: CALL MOTION(#2,-MOT,0,#3,MOT,0,#4
,-MOT,0)
950 Z=Z-50 :: CALL ZEIT(Z):: IF Z<=0 THEN SVAR=-1 :: GOTO 730
960 RETURN
970 FOR I=2 TO 4
980 CALL COINC(#1,#I,16,ST):: IF ST THEN 1010
990 NEXT I
1000 GOTO 720
1010 CALL POSITION(#I,X,Y):: CALL LOCATE(#1,X,Y)
1020 IF I/2-INT(I/2)=0 THEN CALL MOTION(#1,-MOT,0)ELSE CALL MOTION(#1,MOT,0)
1030 CALL POSITION(#1,X,Y):: CALL SOUND(-250,523-X,3):: X=X/8+1 :: Y=Y/8+1 :: CA
```

```
LL GCHAR(X,Y,X1):: CALL GCHAR(X,Y+1,X2):: IF X1=BV OR X2=BV THEN 730
1040 CALL JOYST(1,X,Y):: IF X=0 THEN 1030
1050 CALL POSITION(#1,X1,Y)
1060 IF X=-4 THEN CALL LOCATE(#1,X1,Y-18)ELSE CALL LOCATE(#1,X1,Y+18)
1070 GOTO 720
1080 FOR I=1 TO 4 :: CALL MOTION(#I,0,0):: NEXT I
1090 VWERT=0
1100 CALL POSITION(#1,X,Y):: X=(X+7)/8+1 :: Y=(Y)/8+1
1110 CALL GCHAR(X,Y,X1):: CALL GCHAR(X,Y+1,X2)
1120 IF X1<102 AND X1>95 THEN VWERT=X1
1130 IF X2<102 AND X2>95 THEN VWERT=X2
1140 IF VWERT=0 THEN 1170
1150 VWERT=VWERT-95
1160 ON VWERT GOSUB 1190,1210,1220,1230,1240,1250
1170 CALL MOTION(#2,-MOT,0,#3,MOT,0,#4,-MOT,0)
1180 GOTO 720
1190 CALL HCHAR(X,Y,32,2)
1200 CALL COLOR(13,7,2,12,1,1,14,1,1):: CALL SOUND(-500,-7,0):: BV=128 :: RETURN
1210 CALL HCHAR(X,Y,32,2):: CALL COLOR(12,7,2,13,1,1,14,1,1):: CALL SOUND(-500,-
7,0):: BV=120 :: RETURN
1220 CALL HCHAR(X,Y,32,2):: CALL COLOR(14,7,2,13,1,1,12,1,1):: CALL SOUND(-500,-
7,0):: BV=136 :: RETURN
1230 CALL HCHAR(X,Y,32,2):: WFELD(1)=WFELD(1)+1 :: GOSUB 1260 :: RETURN
1240 CALL HCHAR(X,Y,32,2):: WFELD(2)=WFELD(2)+1 :: GOSUB 1260 :: RETURN
1250 CALL HCHAR(X,Y,32,2):: WFELD(3)=WFELD(3)+1 :: GOSUB 1260 :: RETURN
1260 CALL HCHAR(WFELD(VWERT-3),VWERT,VWERT+36)
1270 P=P+20*VWERT :: CALL PUNKTE(P)
1280 IF BONUS*5000<=P THEN BER=BER+1 :: CALL BAER(BER):: BONUS=BONUS+1
1290 CALL GONG
1300 FOR I=1 TO 3
1310 IF WFELD(I)<2 THEN RETURN
1320 NEXT I
1330 RUNDE=RUNDE+1 :: IF RUNDE>8 THEN RUNDE=8
1340 P=P+Z :: CALL PUNKTE(P):: CALL ZEIT(0):: Z=3500
1350 IF BONUS*5000<=P THEN BER=BER+1 :: CALL BAER(BER):: BONUS=BONUS+1
1360 CALL SONGOFJOY(SF())
1370 CALL DELSPRITE(#1):: CALL HCHAR(1,4,32,3):: CALL HCHAR(2,4,32,3)
1380 CALL COLOR(13,7,2,12,1,1,14,1,1)
1390 GOTO 540
1400 !@P+
1410 SUB SONGOFJOY(B())
1420 GOTO 1430 :: AZ :: I :: J :: K :: M1 :: M2 :: STV :: !@P-
1430 DATA 1,195,246,293,391,2,391,493,1,440,523,1,493,587
1440 DATA 1,184,261,293,369,1,493,587,1,440,523,1,391,493,1,329,440
1450 DATA 1,195,246,293,391,2,246,391,1,261,440,1,293,493
1460 DATA 1,184,261,293,369,2,277,493,2,261,440
1470 DATA 1,195,246,293,391,2,391,493,1,440,523,1,493,587
1480 DATA 1,184,261,293,369,1,493,587,1,440,523,1,391,493,1,329,440
1490 DATA 1,195,246,293,391,2,246,391,1,261,440,1,293,493
1500 DATA 1,184,261,293,369,4,261,440,1,195,246,293,391,4,246,391,0
1510 RESTORE 1430
1520 AZ,STV=0
1530 READ K :: IF K=0 THEN SUBEXIT
1540 READ B(1),B(2),B(3),B(4)
1550 FOR I=1 TO K
1560 FOR J=1 TO 4
1570 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
1580 CALL SOUND(400,M1,1,M2,0,B(J),5)
1590 NEXT J :: NEXT I
1600 GOTO 1530
1610 !@P+
1620 SUBEND
1630 SUB TITEL(A(),KY,B(),SVAR,FARBE)
1640 GOTO 1650 :: AZ :: I :: J :: K :: M1 :: M2 :: S :: STV :: T :: V :: SPI$ ::
 !@P-
1650 CALL MAGNIFY(2):: IF SVAR=0 THEN CALL CLEAR :: CALL SCREEN(2)
1660 DATA L,I,F,T,B,A,E,R
1670 DATA G,A,M,E,O,V,E,R
1680 CALL HCHAR(6,13,114,8):: CALL HCHAR(7,13,122,8):: CALL HCHAR(8,13,129,8)::
CALL HCHAR(9,13,137,8)
1690 RESTORE 1660
1700 IF SVAR THEN RESTORE 1670
1710 IF SVAR=0 THEN DISPLAY AT(18,3):"SPIELANLEITUNG ? <J : N>"
1720 J=1
1730 FOR K=6 TO 8 STEP 2
1740 FOR I=13 TO 19 STEP 2
1750 READ SPI$ :: CALL SPRITE(#J,ASC(SPI$),FARBE,(K-1)*8+1,(I-1)*8+1)
1760 J=J+1
```

```
1770 NEXT I
1780 NEXT K
1790 DISPLAY AT(24,3)SIZE(23):"@ 1984 BY VOLKER BECKER"
1800 DATA 7,10,11,12,3,13,8,14,5
1810 RESTORE 1800
1820 FOR I=1 TO 9
1830 READ A(I)
1840 NEXT I
1850 CALL COLOR(11,A(1),A(2),12,A(3),A(4),13,A(5),A(6),14,A(7),A(8))
1860 RESTORE 1430
1870 AZ,STV=0
1880 READ K :: IF K=0 THEN 1860
1890 READ B(1),B(2),B(3),B(4)
1900 FOR I=1 TO K
1910 FOR J=1 TO 4
1920 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
1930 CALL SOUND(600,M1,1,M2,0,B(J),5):: CALL KEY(KY,T,S):: IF S<>0 THEN 1980
1940 FOR V=9 TO 2 STEP -1 :: A(V)=A(V-1):: NEXT V :: A(1)=A(9)
1950 CALL COLOR(11,A(1),A(2),12,A(3),A(4),13,A(5),A(6),14,A(7),A(8))
1960 NEXT J :: NEXT I
1970 GOTO 1880
1980 CALL DELSPRITE(ALL):: CALL CLEAR
1990 CALL COLOR(11,1,1,12,1,1,13,1,1,14,1,1)
2000 !@P+
2010 SUBEND
2020 SUB PUNKTE(P)
2030 !@P-
2040 DISPLAY AT(1,23)SIZE(5):USING "#####":P :: CALL HCHAR(1,24,38)
2050 !@P+
2060 SUBEND
2070 SUB ZEIT(Z)
2080 !@P-
2090 DISPLAY AT(1,17)SIZE(4):USING "####":Z :: CALL HCHAR(1,18,35)
2100 !@P+
2110 SUBEND
2120 SUB BAER(BER)
2130 !@P-
2140 DISPLAY AT(1,12)SIZE(2):USING "##":BER :: CALL HCHAR(1,13,36)
2150 !@P+
2160 SUBEND
2170 SUB MELODY
2180 GOTO 2190 :: A :: I :: L :: !@P-
2190 DATA 750,195,250,220,500,233,500,195,1000,277
2200 RESTORE 2190
2210 FOR I=1 TO 5 :: READ L,A :: CALL SOUND(L,A,0):: NEXT I
2220 !@P+
2230 SUBEND
2240 SUB GONG
2250 GOTO 2260 :: I :: !@P-
2260 FOR I=0 TO 9 :: CALL SOUND(-150,329,I):: NEXT I
2270 FOR I=0 TO 9 :: CALL SOUND(-150,277,I,329,I+11):: NEXT I
2280 FOR I=0 TO 9 :: CALL SOUND(-150,220,I,277,I+11,329,I+21):: NEXT I
2290 !@P+
2300 SUBEND
2310 SUB EINLEITUNG
2320 GOTO 2330 :: S :: T :: !@P-
2330 CALL CLEAR :: CALL SCREEN(5)
2340 DISPLAY AT(1,1):"DIE MAURER HABEN VERGESSEN  IHR WERKZEUG EIZUSAMMELN UNDDI
ESE AUFGABE MUSS JETZT    DER LIFT BAER UEBERNEHMEN."
2350 DISPLAY AT(5,1):"DAS WERKZEUG IST IN DEN a   VERSTECKT. JE NACH WERKZEUG GI
BT ES :"
2360 DISPLAY AT(8,1):"( =  80 PUNKTE               ) = 100 PUNKTE            *
= 120 PUNKTE"
2370 DISPLAY AT(11,1):"JEDES WERKZEUG IST 2 MAL    VORHANDEN. WIRD JEDOCH EIN  a
 AUFGENOMMEN IN WELCHEM    KEIN WERKZEUG IST SO WERDEN"
2380 DISPLAY AT(15,1):"DIE WAENDE DES NEUBAUES     VERSETZT."
2390 DISPLAY AT(17,1):"LIFT BAER VERLIERT EIN LEBENWENN ER IM FALSCHEN MOMENT  A
US DEM LIFT SPRINGT. EINE  MAUER BERUEHRT ODER DER"
2400 DISPLAY AT(21,1):"ZEITBONUS ABGELAUFEN IST.   BONUSLEBEN ALLE 5000 PUNKTE.N
EUES SPIEL MIT AKTIONSKNOPF"
2410 CALL KEY(1,T,S):: IF T=18 THEN SUBEXIT ELSE 2410
2420 !@P+
2430 SUBEND
2440 SUB ASCIISET
2450 GOTO 2460 :: A$ :: I :: CALL CHAR :: !@P-
2460 DATA 003C666666666663C,0018381818181 83C,003C66060C18307E,003C66061C06663C,00
0C1C2C4C7E0C0C,007E607C0606663C,001C30607C66663C
```

```
2470 DATA 007E060C1830303,003C66663C66663C,003C66663E060C38,C3BD665E5E66BDC3,FFC
3999981999999,FF83C9C9C3C9C983,FFC3999F9F9F99C3
2480 DATA FF83C9C9C9C9C983,FF819F9F839F9F81,FF819F9F839F9F9F,FFC19F9F919999C3,FF
99999981999999,FFC3E7E7E7E7E7C3,FFF9F9F9F9F999C3
2490 DATA FF9993878F879399,FF9F9F9F9F9F9F81,FFBD9981A5999999,FF99898981919199,FF
81999999999981,FF839999839F9F9F,FFC39999999593C9
2500 DATA FF83999983879399,FFC3999FC3F999C3,FF81E7E7E7E7E7E7,FF999999999999C3,FF
999999C3C3E7E7,FF999999A58199DB,FF99DBC3E7C3DB99
2510 DATA FF99DBC3E7E7E7E7,FF81F9F3E7CF9F81,FEFEFE00EFEFEF,FF18FF18FF18FF18,FFFF
FFFF,0000003C40380478,1F306FD0AFA8ABAB,ABABA8AFA0BF80FF
2520 DATA F80CF60BF515D5D5,D5D515F505FD01FF,FEFEFE00EFEFEF,00000442810101,FFFFF
FFF,01820204020201B,000003070D1F0307,0F1F1F0F07040C
2530 DATA 000080C0E0C080C,E0F2F4E4D448C,FEFEFE00EFEFEF,FFFFFFFF,,,0000010307030l
03,074F2F272B1203,0000C0E0B0F8C0E,F0F8F8F0E0203
2540 DATA FEFEFE00EFEFEF,FFFFFFFF,,,000000040C1D373F,3F1D08,0000060F2FEF8F8,FAF
EF0680478
2550 RESTORE 2460
2560 FOR I=48 TO 57 :: READ A$ :: CALL CHAR(I,A$):: NEXT I
2570 FOR I=64 TO 90 :: READ A$ :: CALL CHAR(I,A$):: NEXT I
2580 FOR I=112 TO 143 :: READ A$ :: CALL CHAR(I,A$):: NEXT I
2590 FOR I=96 TO 101 :: CALL CHAR(I,"FFC399F9F3E7FFE7"):: NEXT I
2600 CALL CHAR(35,"3C4281899191523C183418BCBEFE2436"):: CALL CHAR(38,"080C0EFFFF
0E0C08")
2610 CALL CHAR(40,"187EFE7E181818183018 98F87C0E070340E0703C1E1F0F07")
2620 !@P+
2630 SUBEND
```

Luftkampf

für TI-99



Steuern Sie geschickt Ihre Kampfmaschine

mit Ext. Basic-Modul

Ein Spiel für zwei Personen

Zuerst wird das Spielfeld aufgebaut. Die gegnerischen Kampfmaschinen stehen auf ihren Startrampen. Dann starten Sie in den Nachthimmel. Im Luftkampf müssen Sie nun Ihre hartnäckigen Gegner abschütteln. Aber Vorsicht – kommen Sie zu hoch, trudeln Sie ab, und ein Tiefflug durch den Wald führt unweigerlich zur Explosion der Maschine. Die Handhabung Ihrer Kampfmaschine ist relativ einfach. Zu beachten ist, daß der Jet im Steigflug langsam und im Sturzflug schneller wird. Es liegt nun an Ihnen, die richtige Geschwindigkeit zu finden, um im Luftkampf bestehen zu können. Gesteuert wird mit Joystick.

```
100 REM *********************
110 REM *                   *
120 REM *    LUFT KAMPF     *
130 REM *                   *
140 REM *        BY         *
150 REM *                   *
160 REM *    M. STEHLIK     *
220 REM *********************
250 CALL CLEAR
260 DISPLAY AT(12,1):"     L U F T K A M P F":RPT$("-",28):::::
270 FOR I=1 TO 16 :: IF INT(I/2)=I/2 THEN CALL SOUND(125,-1,6)ELSE CALL SOUND(12
5,-3,6)
280 NEXT I
290 PRINT "       FUER 2 SPIELER"
300 PRINT :"SIE STARTEN VON IHRER START-"::"RAMPE IN DEN NACHTHIMMEL UM"
310 PRINT :"IHREN GEGNER ZU BEKAEMPFEN."::"JOYSTICKHEBEL NACH OBEN UND"::
320 PRINT "DAS FLUGZEUG DREHTGEGEN DEN"::"UHRZEIGERSINN, HEBEL NACH"::"UNTEN B
EWIRKT DAS GEGENTEIL."::
330 PRINT "'ALPHA LOCK' LOESEN UND"::"IRGEND EINE TASTE DRUECKEN."
340 CALL KEY(0,K,S):: IF S=0 THEN 340
```

```
350 SC1=0 :: SC2=0
360 DIM G(16):: FOR I=1 TO 16 :: READ G(I):: NEXT I
370 DATA .2,-.2,-.5,-.7,-1,-.7,-.5,-.2,.2,.4,.6,.8,1,.8,.6,.4
380 CALL CLEAR :: CALL SCREEN(2):: CALL MAGNIFY(3)
390 CALL COLOR(3,16,1,4,16,1,6,16,1,2,7,1,5,13,1)
400 DISPLAY AT(12,1):"           ";SC1;":";SC2
410 DIM X(16):: DIM Y(16)
420 FOR I=0 TO 15 :: X(I+1)=COS(I*PI/8):: Y(I+1)=-SIN(I*PI/8):: NEXT I
430 CALL CHAR(76,"0000000000000101",78,"0000000000008080")
440 CALL CHAR(64,"010701070F01070F80E080E0F080E0F801071F3F01073FFF80E0F8FC80E0FC
FF")
450 CALL CHAR(42,"0F0301070101010 1F0C0808080808080")
460 DISPLAY AT(21,1):RPT$("@A",14)
470 DISPLAY AT(22,1):RPT$("BC",14)
480 DISPLAY AT(23,1):RPT$("*+",14)
490 CALL CHAR(40,"FFC1A191898583FFFF83858991A1C1FF")
500 CALL CHAR(68,"FF9FF99FF99FF9FF")
510 CALL HCHAR(24,1,68,32)
520 CALL HCHAR(21,31,41)
530 CALL HCHAR(21,2,40)
540 CALL HCHAR(22,31,40)
550 CALL HCHAR(22,2,41)
560 CALL CHAR(80,"00000000F06070F07060F",82,"00000000000080F080")
570 CALL HCHAR(23,2,40)
580 CALL HCHAR(23,31,41)
590 CALL CHAR(72,"00000000004060307030509",74,"000000004050A0C0C0C060")
600 CALL CHAR(84,"0000000000030E07070F06070C"&RPT$("00",9)&"30C080")
610 CALL CHAR(88,"00000000000002040D1F07070101000000000000000000002040808020C080")
620 CALL CHAR(92,"",93,"0B0F1F1200000000000000000040408080A0E040")
630 CALL CHAR(96,"0000000001010103 0B0F0F09",98,"0000000000000080A0E0E020")
640 CALL CHAR(100,"00000000020201010507030200000000000000000000000080D0F0F848")
650 CALL CHAR(104,"00000000000402010104030100000000000000000004020B0F8E0E08080")
660 CALL CHAR(108,"0000000000000C0301"&RPT$("0",24)&"C070E0E0F060E030")
670 CALL CHAR(112,"00000000000000010F0100000000000000000000000F060E0F0E060F0")
680 CALL CHAR(116,"000000000000000001030C000000000000000000020E060F0E0E070C0")
690 CALL CHAR(120,"00000000010304010204000000000000000000008080E0E0F8B02040")
700 CALL CHAR(124,"000000000203070501010202000000000000000000048F8F0D080")
710 CALL CHAR(128,"0000000004070705010000000000000000000000090F0F0D0C0808080")
720 CALL CHAR(132,"00000000121F0F0B01000000000000000000000040C0E0A080804040")
730 CALL CHAR(136,"000000010107071F0D04020000000000000000000080C02080804020")
740 CALL CHAR(140,"0000000C07060F07070E03"&RPT$("0",24)&"80C030")
750 CALL HCHAR(12,1,32,32):: CALL SPRITE(#1,96,10,149,6,#2,96,8,149,238)
760 P1=5 :: P2=5 :: V1=20 :: V2=20 :: FOR I=1 TO 20 :: CALL HCHAR(1+RND*19,1+RND
*31,76):: NEXT I
770 CALL JOYST(1,A,B):: P1=P1+B/4 :: IF P1>16 THEN P1=P1-16 ELSE IF P1<1 THEN P1
=P1+16
780 V1=V1+2*G(P1):: IF V1>20 THEN V1=20
790 CALL POSITION(#1,D,C):: IF D>149 AND D<195 THEN CALL MOTION(#1,0,0,#2,0,0)::
 CALL COLOR(#1,16):: SC2=SC2+1 :: GOTO 1090
800 IF D>195 OR V1<1 THEN P1=13
810 CALL PATTERN(#1,76+4*P1):: CALL MOTION(#1,V1*Y(P1),V1*X(P1))
820 T=2 :: CALL KEY(1,K,S):: IF K=18 THEN CALL SOUND(100,-1,7,1047,7):: GOTO 890
830 CALL JOYST(2,A,B):: P2=P2+B/4 :: IF P2>16 THEN P2=P2-16 ELSE IF P2<1 THEN P2
=P2+16
840 V2=V2+2*G(P2):: IF V2>20 THEN V2=20
850 CALL POSITION(#2,D,C):: IF D>149 AND D<195 THEN CALL MOTION(#1,0,0,#2,0,0)::
 CALL COLOR(#2,16):: SC1=SC1+1 :: GOTO 1090
860 IF D>195 OR V2<1 THEN P2=13
870 CALL PATTERN(#2,76+4*P2):: CALL MOTION(#2,V2*Y(P2),V2*X(P2))
880 T=1 :: CALL KEY(2,K,S):: IF K=18 THEN CALL SOUND(100,1047,7,-1,7):: GOTO 990
 ELSE 770
890 SC1=SC1+1 :: V1=V1-2 :: CALL POSITION(#1,D,C):: CALL SPRITE(#3,76,16,D,C,80*
Y(P1),80*X(P1))
900 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
910 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
920 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
930 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
940 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
950 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
960 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
970 CALL COINC(#3,#2,3,H):: IF H=-1 THEN 1090
980 CALL DELSPRITE(#3):: SC1=SC1-1 :: GOTO 830
990 SC2=SC2+1 :: V2=V2-2 :: CALL POSITION(#2,D,C):: CALL SPRITE(#3,76,16,D,C,80*
Y(P2),80*X(P2))
1000 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1010 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1020 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1030 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
```

```
1040 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1050 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1060 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1070 CALL COINC(#3,#1,3,H):: IF H=-1 THEN 1090
1080 CALL DELSPRITE(#3):: SC2=SC2-1 :: GOTO 770
1090 CALL SOUND(500,-7,0,110,0,111,0):: CALL DELSPRITE(#3)
1100 CALL PATTERN(#T,72):: CALL SOUND(500,-6,2,110,2,117,2)
1110 CALL CHAR(72,"00000000000408050A0A",70,"0000000402010A0500090"):: CALL SOU
ND(500,-5,4,110,4,117,4)
1120 CALL DELSPRITE(ALL):: GOTO 380
```

Clipper Flipper

für Laser 2001

Es ist schwer, in BASIC Spielprogramme zu schreiben, welche eine relativ gute Grafik bringen und auch noch in einer annehmbaren Geschwindigkeit laufen. Beim Spiel CLIPPER FLIPPER ist dies jedoch gelungen.

CLIPPER FLIPPER ist, wie der Name schon vermuten läßt, eine Flipper-Variante. Doch ist es bei CLIPPER FLIPPER möglich, die Laufbahn der Kugel im voraus zu sehen und diese Laufbahn auch in einem gewissen Rahmen zu beeinflussen.

CLIPPER FLIPPER ist auf dem LASER 2001 in LASERBASIC geschrieben und ist auf der Konsole allein, auch ohne 16-kB-Memory-Expansion, zu spielen. Der Speicher der Konsole reicht auch noch bei angeschlossenem Diskettenlaufwerk aus.

CLIPPER FLIPPER wird mittels linkem Joystick gesteuert, wobei jeder handelsübliche Joystick benutzt werden kann (z. B. ATARI oder SPECTRAVIDEO).

Doch nun zum Spiel: Tippen Sie das Programm ein oder lassen Sie dieses von Cassette bzw. Diskette ein. Starten Sie das Programm jetzt mit RUN, und das Titel-Bild erscheint.

Durch Druck auf den roten Aktionsknopf des linken Joysticks (bei LASER Joysticks linker Aktionsknopf) verlassen Sie das Titelbild. Der Computer baut jetzt das Spielfeld auf.

Sie sehen auf dem Bildschirm 14 Bahnen, in welchen die Kugel rollen kann. Zur Vereinfachung der Spielerklärung numerieren wir die 14 Bahnen wie folgt durch: Bahn 1 ist die linkeste Bahn auf dem Bildschirm. Bahn 13 ist die zweite Bahn von rechts und gleichzeitig die Vorratsbahn der drei Kugeln. Bahn 14 ist die rechte Bahn und gleichzeitig die Einwurfbahn.

Die Bahnen numerieren wir von links nach rechts fortlaufend durch.

Ferner sind im Spielfeld 30 Clipper eingebaut, die je nach Richtung die Kugel in eine andere Bahn umlenken oder von der Kugel umgeklappt werden.

Die blauen Clipper lenken die Kugel um, wenn diese von oben nach unten über den Bildschirm rollt. Rollt die Kugel jedoch von unten nach oben über den Bildschirm, so werden die blauen Clipper umgeklappt.

Die roten Clipper werden von der Kugel umgeklappt, wenn diese von oben nach unten über den Bildschirm rollt. Rollt die Kugel jedoch von unten nach oben über den Bildschirm, so wird die Kugel, sofern sich ein entsprechender roter Clipper in der Bahn befindet, in die andere Bahm umgelenkt. Die Kugel rollt dann jedoch nicht weiter nach oben, sondern rollt wieder über den Bildschirm nach unten.

Um eine Kugel einzuwerfen, ziehen Sie den Joystick nach unten. Die Einwurffeder spannt sich und die Kugel fällt in Bahn 14 über die Einwurffeder. Lassen Sie den Joystick jetzt los, und die Einwurffeder schnellt empor und bringt die Kugel in Bewegung. Die Kugel fällt nun in irgendeine Bahn zwischen 1 und 13. Softwaremäßig ist Bahn 14 eine „Einbahnstraße" für die Kugel. Sobald die Kugel in Bahn 13 fällt, kann eine neue Kugel eingeworfen werden. Sie haben so praktisch eine Freikugel gewonnen.

Unten im Bild ist ein kleiner blauer FLIPPER eingeblendet. Dieser FLIPPER kann mittels Joystick nach rechts und links bewegt werden. Der Flipper kann zwischen den Bahnen 1 und 12 bewegt werden. Wenn die Kugel am unteren Bildschirmrand eine Bahn verläßt und auf den FLIPPER stößt, so wird sie in die Bahn, aus der sie gekommen ist, zurückgeschlagen. Rollt die Kugel jedoch aus der Bahn, ohne auf den Flipper zu treffen, so ist sie verloren.

Wenn die Kugel den oberen Bildschirmrand wieder erreicht, so rollt sie dort entlang und fällt wieder in eine der Bahnen 1 bis 13.

Wenn Ihnen die Stellung der Clipper nicht gefällt, so können Sie diese verändern, indem Sie den Aktionsknopf drücken, sobald die Kugel den FLIPPER berührt. Die Kugel verharrt dann in dieser Stellung, und alle CLIPPER kippen um. Danach wird die Kugel vom FLIPPER wieder zurückgeworfen.

In den einzelnen Bahnen sind kleine violette Kontakte angebracht. Jedes Mal, wenn die Kugel an einem Kontakt vorbeiläuft, wird dies durch ein kurzes akustisches Signal angezeigt, und Ihrem Punktekonto werden 5 Punkte gutgeschrieben. Die Punkte werden jedoch immer nur dann angezeigt, wenn die Kugel ins AUS geht oder wenn die Kugel in Bahn 13 rollt. Dies geschieht, um den Spielablauf nicht unnötig zu verlangsamen.

Wenn alle 3 Kugeln ins AUS gegangen sind, so ist das Spiel beendet, und die Schrift TILT erscheint auf dem Bildschirm.

Um ein neues Spiel zu beginnen, drücken Sie den Aktionsknopf am Joystick.

```
10  REM *******************************
12  REM *       CLIPPER FLIPPER       *
14  REM *-----------------------------*
16  REM * LASER 2001 + li. Joystick   *
18  REM * geschrieben in LASERBASIC   *
20  REM *-----------------------------*
22  REM * (c) 1984 by Tronicsoft      *
24  REM               V.Becker        *
26  REM *******************************
28  REM
30  REM
100  DIM P%(5,6),B%(3,2),R%(5,6,2)
110  FOR I = 1 TO 5: FOR K = 1 TO 6:P%(I,K) = 0: NEXT K: NEXT I
120 B%(1,1) = 28:B%(1,2) = 19:B%(2,1) = 27:B%(2,2) = 19:B%(3,1) = 27:B%(3,2) = 1
8
122  GOSUB 25000
125  GOSUB 41000
```

```
130  GOSUB 50000
135 PK = 0
140  GOSUB 31000
145  CALL  - 1536
150  IF  PEEK (24) = 2 THEN 49000
160  GOTO 145
170  GOSUB 31000
180  COLOR= 2,1
190  FOR I = 72 TO 112: UNPLOT 40,I TO 223,I
195  SOUND (272 - I,1,15),(273 - I,1,15),(271 - I,1,15)
200  NEXT I
205 C = 2
210 L = 1:Z = 82
215  COLOR= C,1:C = C + 1: IF C > 13 THEN C = 2
220  FOR S = 48 TO 208 STEP 48
230  FOR I = 0 TO 5
240  ON L GOSUB 40700,40200,40100,40700
250  NEXT I
260  CALL  - 1536: IF  PEEK (24) > 100 THEN 300
270 L = L + 1
280  NEXT S
290  GOTO 210
300 B%(1,1) = 28:B%(1,2) = 19:B%(2,1) = 27:B%(2,2) = 19:B%(3,1) = 27:B%(3,2) = 1
8
310  GOTO 130
1000 BS = E:BZ = 1
1010 S = BS:Z = BZ: GOSUB 51100
1015  IF BZ = 22 THEN 11000
1016  IF BS = 27 AND BZ = 18 THEN  GOSUB 50260: GOTO 140
1017  CALL  - 1536: IF  PEEK (24) = 0 THEN 1020
1018  IF  PEEK (24) > 100 THEN 1020
1019  GOSUB 12000
1020 BZ = BZ + 1: IF ((BZ + 1) / 4) =  INT ((BZ + 1) / 4) THEN 5000
1030 S = BS:Z = BZ: GOSUB 51000
1035  IF BZ > 4 AND BZ < 20 THEN  IF ((BZ - 2) / 4) -  INT ((BZ - 2) / 4) = 0 TH
EN  GOSUB 32000
1040  GOTO 1010
2000 S = BS:Z = BZ: GOSUB 51100
2015  IF BZ = 1 THEN 9600
2016  IF BS = 27 AND BZ = 18 THEN 150
2017  CALL  - 1536: IF  PEEK (24) = 0 THEN 2020
2018  IF  PEEK (24) > 100 THEN 2020
2019  GOSUB 12000
2020 BZ = BZ - 1: IF (BZ / 4) =  INT (BZ / 4) THEN 15000
2030 S = BS:Z = BZ: GOSUB 51000
2035  IF BZ > 4 AND BZ < 20 THEN  IF ((BZ - 2) / 4) -  INT ((BZ - 2) / 4) = 0 TH
EN  GOSUB 32000
2040  GOTO 2000
5000 ID = (BZ + 1) / 4
5010  IF (ID / 2) =  INT (ID / 2) THEN 6000
5015 ET = (BS - 2) / 4 -  INT ((BS - 2) / 4)
5020 I2 =  INT ((BS - 2) / 4) + 1
5030  GOTO 7000
6000 I2 =  INT ((BS - 4) / 4) + 1
6005 ET = (BS - 4) / 4 -  INT ((BS - 4) / 4)
6010  GOTO 7000
7000  IF I2 > 6 THEN 1030
7005 SK = P%(ID,I2)
7010  ON SK GOTO 8000,8500,9000,9500
8000  IF ET = .25 THEN 1030
8005  IF BS = 3 THEN 1030
8010 S = BS:Z = BZ: GOSUB 51000
8012 S = BS:Z = BZ: GOSUB 51100
8015 BS = BS - 1
8020 S = BS:Z = BZ: GOSUB 51000
8030 S = BS:Z = BZ: GOSUB 51100
8040 BS = BS - 1
```

```
8050  GOTO 1030
8500  IF ET = .25 THEN 1030
8510 S = R%(ID,I2,1):Z = R%(ID,I2,2):J = ID:I = I2: GOSUB 55300
8520  GOTO 1030
9000  IF ET = .75 THEN 1030
9005  IF BS = 27 THEN 1030
9010 S = BS:Z = BZ: GOSUB 51000
9012 S = BS:Z = BZ: GOSUB 51100
9015 BS = BS + 1
9020 S = BS:Z = BZ: GOSUB 51000
9030 S = BS:Z = BZ: GOSUB 51100
9040 BS = BS + 1
9050  GOTO 1030
9500  IF ET = .75 THEN 1030
9510 S = R%(ID,I2,1):Z = R%(ID,I2,2):J = ID:I = I2: GOSUB 55100
9520  GOTO 1030
9600 E =  INT ( RND (255) * 27)
9610  IF E < 3 THEN E = 3
9620  IF  INT (E / 2) = E / 2 THEN E = E + 1
9630  IF E > 27 THEN E = 27
9640  FOR I = BS TO E STEP  SGN (E - BS)
9650 S = I:Z = 1: GOSUB 51000
9660 S = I:Z = 1: GOSUB 51100
9670  NEXT I
9680 S = E:Z = 1: GOSUB 51000
9690  GOTO 1000
10000  FOR J = 1 TO 5: FOR I = 1 TO 6
10010 S = R%(J,I,1):Z = R%(J,I,2)
10020 SK = P%(J,I)
10030  ON SK GOSUB 55200,55300,55000,55100
10040  NEXT I: NEXT J
10050  RETURN
11000  IF BS = PF THEN 11500
11005  GOSUB 60000
11010  FOR I = 3 TO 1 STEP  - 1
11020  IF B%(I,1) <  > 0 THEN B%(I,1) = 0: GOTO 11100
11030  NEXT I
11100  IF B%(1,1) = 0 THEN 170
11101 ZI = PF:PF = BS: GOSUB 14200:PF = ZI
11105  GOSUB 50260
11110  GOTO 140
11500 S = BS:Z = BZ: GOSUB 51000
11510  CALL  - 1536: IF  PEEK (24) > 100 THEN  GOSUB 10000
11515  SOUND (135,1,15)
11520  GOTO 2000
12000 JS =  PEEK (24) / 2
12010  IF JS > 10 THEN 12500
12020  ON JS GOTO 12999,12030,12030,12999,12999,12999
12030  IF PF = 25 THEN 12999
12040  GOSUB 14200:PF = PF + 2: GOSUB 14100
12050  RETURN
12500 JS = JS - 15
12510  ON JS GOTO 12520,12520,12999,12999,12520
12520  IF PF = 3 THEN 12999
12530  GOSUB 14200:PF = PF - 2: GOSUB 14100
12540  RETURN
12999  RETURN
14000  COLOR= 1,1
14010  FOR I = 3 TO 25 STEP 2
14020 S = 8 * I
14030  PLOT S,180 TO S + 1,180
14040  PLOT S + 6,180 TO S + 7,180
14050  PLOT S,181 TO S + 2,181
14060  PLOT S + 5,181 TO S + 7,181
14070  PLOT S,182 TO S + 7,182
14080  NEXT I
14082 PF = 15: GOSUB 14100
```

```
14090  RETURN
14100  COLOR= 7,1
14110  PLOT PF * 8,180 TO PF * 8,182
14120  RETURN
14200  COLOR= 1,1
14210  PLOT PF * 8,180 TO PF * 8,182
14220  RETURN
15000 ID = BZ / 4
15010  IF (ID / 2) =  INT (ID / 2) THEN 16000
15015 ET = (BS - 2) / 4 -  INT ((BS - 2) / 4)
15020 I2 =  INT ((BS - 2) / 4) + 1
15030  GOTO 17000
16000 I2 =  INT ((BS - 4) / 4) + 1
16005 ET = (BS - 4) / 4 -  INT ((BS - 4) / 4)
16010  GOTO 17000
17000  IF I2 > 6 THEN 2030
17005 SK = P%(ID,I2)
17010  ON SK GOTO 18500,18000,19500,19000
18000  IF ET = .25 THEN 2030
18005  IF BS = 3 THEN 2030
18010 S = BS:Z = BZ: GOSUB 51000
18012 S = BS:Z = BZ: GOSUB 51100
18015 BS = BS - 1
18020 S = BS:Z = BZ: GOSUB 51000
18030 S = BS:Z = BZ: GOSUB 51100
18040 BS = BS - 1
18050  GOTO 1030
18500  IF ET = .25 THEN 2030
18510 S = R%(ID,I2,1):Z = R%(ID,I2,2):J = ID:I = I2: GOSUB 55200
18520  GOTO 2030
19000  IF ET = .75 THEN 2030
19005  IF BS = 27 THEN 2030
19010 S = BS:Z = BZ: GOSUB 51000
19012 S = BS:Z = BZ: GOSUB 51100
19015 BS = BS + 1
19020 S = BS:Z = BZ: GOSUB 51000
19030 S = BS:Z = BZ: GOSUB 51100
19040 BS = BS + 1
19050  GOTO 1030
19500  IF ET = .75 THEN 2030
19510 S = R%(ID,I2,1):Z = R%(ID,I2,2):J = ID:I = I2: GOSUB 55000
19520  GOTO 2030
20000  UNPLOT 95,185 TO 98,185
20010  UNPLOT 94,186 TO 94,189
20020  UNPLOT 99,186 TO 99,189
20030  UNPLOT 95,190 TO 98,190
20040  UNPLOT 96,186 TO 97,186
20050  UNPLOT 96,189 TO 97,189
20060  UNPLOT 95,187 TO 95,188
20070  UNPLOT 102,185 TO 102,190
20080  UNPLOT 101,186
20090  UNPLOT 104,190 TO 107,190 TO 107,185 TO 104,185 TO 104,188 TO 107,188
20100  UNPLOT 109,185 TO 113,185 TO 113,190 TO 109,190 TO 109,185
20110  UNPLOT 109,187 TO 113,187
20120  UNPLOT 115,185 TO 115,188 TO 119,188
20130  UNPLOT 118,185 TO 118,190
20140  UNPLOT 121,185 TO 138,185
20150  UNPLOT 123,186 TO 123,190
20160  UNPLOT 125,187 TO 125,190
20170  UNPLOT 126,188
20171  UNPLOT 140,185
20172  UNPLOT 127,187 TO 128,187
20180  UNPLOT 130,187 TO 133,187 TO 133,190 TO 130,190 TO 130,187
20190  UNPLOT 135,187 TO 135,190
20200  UNPLOT 136,188
20210  UNPLOT 137,187 TO 138,187 TO 138,190
20220  UNPLOT 140,187 TO 140,190
```

```
20230  UNPLOT 144,187 TO 142,187 TO 142,190 TO 144,190
20240  UNPLOT 142,185 TO 144,185
20250  UNPLOT 157,185 TO 147,185 TO 147,188 TO 151,188 TO 151,190 TO 147,190
20260  UNPLOT 153,187 TO 156,187 TO 156,190 TO 153,190 TO 153,187
20270  UNPLOT 158,186 TO 158,190
20280  UNPLOT 160,185 TO 159,185
20290  UNPLOT 159,187
20300  UNPLOT 162,185 TO 162,190
20310  UNPLOT 161,187 TO 163,187
20320  RETURN
25000  GR
25005  COLOR= 8,1
25010  FOR I = 0 TO 7: RECT (I,I),(255 - I,191 - I): NEXT I
25020  FOR I = 8 TO 15: PLOT I,8 TO I,183
25030  PLOT 255 - I,8 TO 255 - I,183
25040  NEXT I
25045  GOSUB 20000
25050  RETURN
30000  UNPLOT S + 1,1 TO S + 5,1 TO S + 5,6 TO S + 1,6 TO S + 1,1
30010  RETURN
30100  UNPLOT S + 4,2 TO S + 5,1 TO S + 5,6
30110  RETURN
30200  UNPLOT S + 1,1 TO S + 5,1 TO S + 5,4 TO S + 1,4 TO S + 1,6 TO S + 5,6
30210  RETURN
30300  UNPLOT S + 1,1 TO S + 5,1 TO S + 5,6 TO S + 1,6
30310  UNPLOT S + 3,3 TO S + 5,3
30320  RETURN
30400  UNPLOT S + 1,1 TO S + 1,3 TO S + 5,3
30410  UNPLOT S + 4,1 TO S + 4,6
30420  RETURN
30500  UNPLOT S + 5,1 TO S + 1,1 TO S + 1,3 TO S + 5,3 TO S + 5,6 TO S + 1,6
30510  RETURN
30600  UNPLOT S + 5,1 TO S + 1,1 TO S + 1,6 TO S + 5,6 TO S + 5,3 TO S + 2,3
30610  RETURN
30700  UNPLOT S + 1,1 TO S + 5,1 TO S + 5,6
30710  RETURN
30800  UNPLOT S + 1,1 TO S + 5,1 TO S + 5,6 TO S + 1,6 TO S + 1,1
30810  UNPLOT S + 1,3 TO S + 5,3
30820  RETURN
30900  UNPLOT S + 1,6 TO S + 5,6 TO S + 5,1 TO S + 1,1 TO S + 1,4 TO S + 5,4
30910  RETURN
31000  COLOR= 10,1
31010 P$ =  STR$ (PK)
31020 S = 240 - 7 *  LEN (P$)
31030 L =  VAL ( MID$ (P$,1,1)) + 1
31040  FOR I = 1 TO 6: PLOT S,I TO S + 6,I
31050  NEXT I
31060  ON L GOSUB 30000,30100,30200,30300,30400,30500,30600,30700,30800,30900
31070  IF  LEN (P$) <  = 1 THEN  RETURN
31080 P$ =  RIGHT$ (P$, LEN (P$) - 1)
31090  GOTO 31020
32000 PK = PK + 5
32010  SOUND (50,1,15)
32050  RETURN
40000  PLOT S + 18 + I,Z + 5 - I TO S + I,Z + 5 - I TO S + I,Z + 23 - I TO S + 1
8 + I,Z + 23 - I
40010  RETURN
40100  PLOT S + I,Z + 5 - I TO S + I,Z + 23 - I TO S + 18 + I,Z + 23 - I
40110  RETURN
40140  ON L GOSUB 40000,40100,40200,40300,40300,40400,40500
40200  PLOT S + 9 + I,Z + 5 - I TO S + 9 + I,Z + 23 - I
40210  RETURN
40300  PLOT S + I,Z + 23 - I TO S + I,Z + 5 - I TO S + 18 + I,Z + 5 - I TO S + 1
8 + I,Z + 14 - I TO S + 3 + I,Z + 14 - I
40310  RETURN
40400  GOSUB 40600
40410  PLOT S + I,Z + 23 - I TO S + 18 + I,Z + 23 - I
```

```
40420  RETURN
40500  PLOT S + I,Z + 23 - I TO S + I,Z + 5 - I TO S + 18 + I,Z + 5 - I TO S + 1
8 + I,Z + 14 - I TO S + 3 + I,Z + 14 - I TO S + 18 + I,Z + 23 - I
40510  RETURN
40600  PLOT S + 18 + I,Z + 5 - I TO S + I,Z + 5 - I TO S + I,Z + 23 - I
40610  PLOT S + I,Z + 14 - I TO S + 12 + I,Z + 14 - I
40620  RETURN
40700  GOSUB 40200
40710  PLOT S + I,Z + 5 - I TO S + 18 + I,Z + 5 - I
40720  RETURN
41000 C = 2
41010 L = 1:Z = 64
41015  COLOR= C,1:C = C + 1: IF C > 13 THEN C = 2
41020  FOR S = 48 TO 208 STEP 24
41030  FOR I = 0 TO 2
41040  ON L GOSUB 40000,40100,40200,40300,40300,40400,40500
41050  NEXT I
41060  CALL  - 1536: IF  PEEK (24) > 100 THEN  RETURN
41070 L = L + 1
41080  NEXT S
41090 Z = 104:L = 1
41115  COLOR= C,1:C = C + 1: IF C > 13 THEN C = 2
41120  FOR S = 48 TO 208 STEP 24
41130  FOR I = 0 TO 2
41140  ON L GOSUB 40600,40100,40200,40300,40300,40400,40500
41150  NEXT I
41160  CALL  - 1536: IF  PEEK (24) > 100 THEN  RETURN
41170 L = L + 1
41180  NEXT S
41185  FOR K = 1 TO 4
41190  FOR I = 140 TO 80 STEP  - 20
41200  SOUND (I,1,15),(I + 2,1,15),(I - 2,1,15)
41210  NEXT I
41215  NEXT K
41220  GOTO 41010
49000 VG = B%(1,1)
49005  IF VG = 0 THEN 170
49010  COLOR= 13,1
49020  FOR I = 140 TO 159: UNPLOT 232,I TO 239,I
49030  NEXT I
49040  FOR I = 160 TO 163: PLOT 232,I TO 239,I
49050  NEXT I
49055  SOUND (50,1,10),(60,2,10),(70,3,10),(1,4,10,1)
49060  FOR I = 3 TO 1 STEP  - 1
49070  IF B%(I,1) <  > 0 THEN 49100
49080  NEXT I
49090  GOTO 170
49100 S = B%(I,1):Z = B%(I,2): GOSUB 51100
49110 S = 29:Z = 19: GOSUB 51000
49120  CALL  - 1536
49130  IF  PEEK (24) = 0 THEN 49150
49140  GOTO 49120
49150  COLOR= 13,1
49155  FOR I = 160 TO 163: UNPLOT 232,I TO 239,I
49156  NEXT I
49160  FOR I = 140 TO 144: PLOT 232,I TO 239,I
49170  NEXT I
49180  PLOT 235,145 TO 235,184 TO 236,184 TO 236,145
49185  SOUND (1,1,15),(2,1,15),(3,1,15),(2,1,15,2)
49190  FOR I = 16 TO 1 STEP  - 1
49200 S = 29:Z = I: GOSUB 51000
49210 S = 29:Z = I: GOSUB 51100
49220  NEXT I
49230 E =  INT ( RND (255) * 27)
49240  IF E < 3 THEN E = 3
49250  IF  INT (E / 2) = E / 2 THEN E = E + 1
49260  IF E > 27 THEN E = 27
```

```
49270  FOR I = 29 TO E STEP  - 1
49280 S = I:Z = 1: GOSUB 51000
49290 S = I:Z = 1: GOSUB 51100
49300  NEXT I
49320 S = E:Z = 1: GOSUB 51000
49330  GOTO 1000
50000  GR
50005  COLOR= 10,1
50010  FOR I = 0 TO 7: RECT (I,I),(255 - I,191 - I): NEXT I
50020  FOR I = 8 TO 15: PLOT I,8 TO I,183
50030  PLOT 255 - I,8 TO 255 - I,183
50040  NEXT I
50045  GOSUB 20000
50050  COLOR= 2,1
50054  RESTORE
50055  FOR I = 1 TO 5: FOR K = 1 TO 6: READ A,B: NEXT K: NEXT I
50060  FOR RD = 1 TO 44
50070  READ S,Z,L:S = S * 8:Z = Z * 8:L = L * 8: GOSUB 54000: NEXT RD
50080  FOR I = 160 TO 183: PLOT 208,I TO 231,I
50090  NEXT I
50094  RESTORE
50100  FOR J = 1 TO 5: FOR I = 1 TO 6
50110  READ S,Z
50115 R%(J,I,1) = S:R%(J,I,2) = Z
50120 X =  INT ( RND (255) * 3.5 + 1)
50130  ON X GOSUB 55000,55100,55200,55300
54000  FOR J = 0 TO 1
54010  PLOT S + J,Z + 3 TO S + 3 + J,Z TO S + 4 - J,Z TO S + 7 - J,Z + 4 TO S +
7 - J,Z + 4 + L TO S + 4 - J,Z + L + 7 TO S + 3 + J,Z + 7 + L TO S + J,Z + 3 + L
 TO S + J,Z + 3
54020  NEXT J
54030  PLOT S + 3,Z TO S + 3,Z + 7 + L TO S + 4,Z + 7 + L TO S + 4,Z
54040  RETURN
55000  COLOR= 5,1
55010 S = 8 * S + 3:Z = 8 * Z + 7
55015  UNPLOT S,Z TO S - 11,Z - 7 TO S + 1,Z
55020  PLOT S,Z TO S + 12,Z - 7 TO S + 1,Z
55022  SOUND (20,1,15),(40,1,15),(60,1,15),(2,1,15,3)
55025 P%(J,I) = 1
55030  RETURN
55100  COLOR= 8,1
55110 S = 8 * S + 3:Z = (Z - 1) * 8
55115  UNPLOT S,Z TO S - 11,Z + 7 TO S + 1,Z
55120  PLOT S,Z TO S + 12,Z + 7 TO S + 1,Z
55122  SOUND (20,1,15),(40,1,15),(60,1,15),(2,1,15,3)
55125 P%(J,I) = 2
55130  RETURN
55200  COLOR= 5,1
55210 S = 8 * S + 3:Z = 8 * Z + 7
55220  UNPLOT S,Z TO S + 12,Z - 7 TO S + 1,Z
55222  SOUND (20,1,15),(40,1,15),(60,1,15),(2,1,15,3)
55223  PLOT S,Z TO S - 11,Z - 7 TO S + 1,Z
55225 P%(J,I) = 3
55230  RETURN
55300  COLOR= 8,1
55310 S = 8 * S + 3:Z = (Z - 1) * 8
55320  UNPLOT S,Z TO S + 12,Z + 7 TO S + 1,Z
55324  PLOT S,Z TO S - 11,Z + 7 TO S + 1,Z
55325 P%(J,I) = 4
55326  SOUND (20,1,15),(40,1,15),(60,1,15),(2,1,15,3)
55330  RETURN
60000  SOUND (255,1,0),(255,1,0),(255,1,0),(1,1,0,1)
60001  SGEN 159,191,223,255
60002  SOUND (120,1,0),(150,1,0)
60005  FOR I = 1 TO 10
60010  SGEN 50 + 20 * I,I,20 + 10 * I
60015  FOR K = 1 TO 5 * I: NEXT K
```

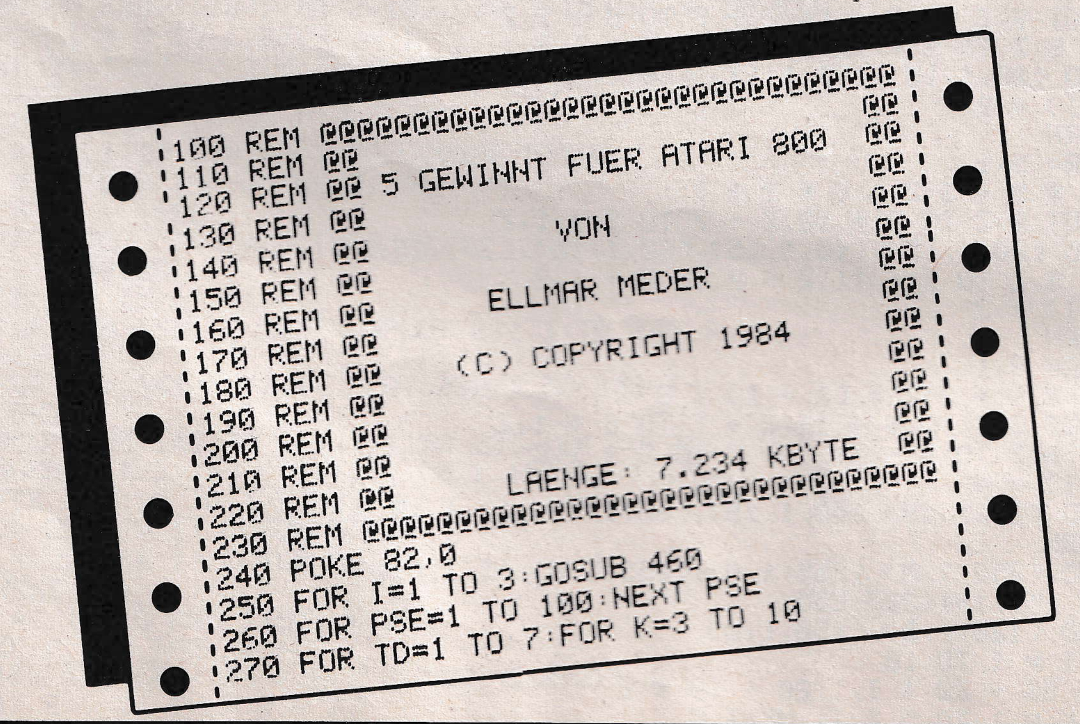
```
60020  NEXT I
60025  SOUND (255,1,0),(255,1,0),(255,1,0),(1,1,0,1)
60030  RETURN
62040  DATA  4,4,8,4,12,4,16,4,20,4,24,4,6,8,10,8,14,8,18,8,22,8,26,8,4,12,8,12,
12,12,16,12,20,12,24,12
62050  DATA  6,16,10,16,14,16,18,16,22,16,26,16,4,20,8,20,12,20,16,20,20,20,24,2
0
63000  DATA   2,1,21,4,2,0,4,5,5,4,13,5,4,21,0,6,2,4,6,9,5,6,17,4,8,2,0,8,5,5,8,
13,5,8,21,0
63010  DATA  10,2,4,10,9,5,10,17,4,12,2,0,12,5,5,12,13,5,12,21,0,14,2,4,14,9,5,1
4,17,4
63020  DATA   16,2,0,16,5,5,16,13,5,16,21,0,18,2,4,18,9,5,18,17,4,20,2,0,20,5,5,
20,13,5,20,21,0
63030  DATA  22,2,4,22,9,5,22,17,4,24,2,0,24,5,5,24,13,5,24,21,0,26,2,4,26,9,5,2
6,17,5,28,2,16
```



Fünf gewinnt ist ein Strategiespiel für 2 Personen mit 2 Joysticks, geschrieben in ATARI Basic.

Taktisches Gespür, strategisches und logisches Denken, schnelles Erkennen der Situationen sind gefragt. – Eine Version des beliebten Strategiespieles. Zu Beginn des Spieles wird eine Spielanleitung angeboten, die das Programm kurz beschreibt.

Der Sinn des Spieles ist relativ einfach: Wer es als erster schafft, 5 Steine in eine Reihe zu bekommen, horizontal, vertikal oder diagonal, gewinnt das Spiel.
Für Spieler 1 (blau) ist am Anfang des Spieles der 1. Stein gesetzt.
Nach Ende des Spieles wird der Verlierer getröstet und durch drücken der Start-Taste das Spiel von neuem begonnen.
Also dann: Viel Spaß!



```
100 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
110 REM @@                              @@
120 REM @@ 5 GEWINNT FUER ATARI 800      @@
130 REM @@                              @@
140 REM @@            VON               @@
150 REM @@                              @@
160 REM @@       ELLMAR MEDER           @@
170 REM @@                              @@
180 REM @@     (C) COPYRIGHT 1984       @@
190 REM @@                              @@
200 REM @@                              @@
210 REM @@                              @@
220 REM @@        LAENGE: 7.234 KBYTE   @@
230 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
240 POKE 82,0
250 FOR I=1 TO 3:GOSUB 460
260 FOR PSE=1 TO 100:NEXT PSE
270 FOR TD=1 TO 7:FOR K=3 TO 10
```

```
280 SOUND 1,K,10,8
290 NEXT K:NEXT TD
300 NEXT I
310 SOUND 1,0,0,0
320 GOTO 450
330 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
340 REM @@ UNTERPROGRAM ZUM FEST-     @@
350 REM @@ STELLEN OB KONSOLEN-      @@
360 REM @@ TASTE GEDRUECKT           @@
370 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
380 POKE 53279,8
390 TRAP 390:IF PEEK(53279)<>6 OR PEEK(53279)=7 THEN GOTO 390
400 RETURN
410 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
420 REM @@ UNTERPROGRAMM ZUR DAR-    @@
430 REM @@ STELLUNG DES P.ANFANGS    @@
440 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
450 POKE 755,1:GOTO 600
460 GRAPHICS 5
470 SETCOLOR 4,0,0:SETCOLOR 2,0,0
480 SETCOLOR 1,13,9:COLOR 2
490 PLOT 50,8:DRAWTO 30,8
500 DRAWTO 30,19:DRAWTO 47,19
510 DRAWTO 47,28:DRAWTO 30,28
520 DRAWTO 30,31:DRAWTO 50,31
530 DRAWTO 50,16:PLOT 50,16
540 DRAWTO 33,16:DRAWTO 33,11
550 DRAWTO 50,11:DRAWTO 50,8
560 PLOT 0,0:DRAWTO 79,0
570 DRAWTO 79,39:DRAWTO 0,39
580 DRAWTO 0,0
590 RETURN
600 PRINT "(12 SPACES) 5 G E W I N N T"          ";
610 PRINT "(18 SPACES)       VON(20 SPACES)
620 PRINT "(13 SPACES)  ELLMAR MEDER";
630 PRINT "(25 SPACES)            (C) 1984";
640 FOR PSE=1 TO 1000:NEXT PSE
650 DIM ANTW$(1)
660 PRINT "}Wird Spielanleitung benoetigt (J/N) ";
670 INPUT ANTW$:IF ANTW$="N" THEN GOTO 860
680 GRAPHICS 0:SETCOLOR 2,0,0
720 PRINT "Dies ist ein Spiel fuer 2 Personen.":PRINT
730 PRINT "Der Sinn des Spieles besteht darin,dass":PRINT
740 PRINT "man versuchen muss 5 Steine entweder in":PRINT
750 PRINT " HORIZONTALER, VERTIKALER  ODER IN DIA-":PRINT
760 PRINT "GONALER Reihenfolge zusammenzusetzten.":PRINT
770 PRINT "Durch druecken des roten Knopfes am":PRINT
780 PRINT " Joystick(1/2) wird der Stein gesetzt.":PRINT
790 PRINT "Fuer Spieler 1 (blau) ist der erste ":PRINT
800 PRINT "Stein zu Beginn des Spieles gesetzt.":PRINT
810 PRINT "VIEL SPASS                  --->START";
820 GOSUB 380
830 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
840 REM @@BEGINN DES HAUPTPROGRAMMS  @@
850 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
860 CLR :POKE 12,119:POKE 13,228:POKE 755,1:POKE 82,0
870 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
880 REM @@@     INITIALISIERUNG     @@@
890 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
900 GOSUB 1800
910 PLR=1-PLR
920 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

```
930 REM @@ FESTSTELLUNG DER GEG-       @@
940 REM @@ WERTIGEN POSITION           @@
950 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
960 PK=PEEK(PPOS(PLR))
970 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
980 REM @@  SETZEN DER STEINE          @@
990 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1000 POKE PPOS(PLR),PIECE(PLR)
1010 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1020 REM @ BLINKDAUER DES STEINCHENS @@
1030 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1040 FOR BLD=1 TO 10:NEXT BLD
1050 ST=STICK(PLR):IF STRIG(PLR)=0 AND PK=83 THEN 1170
1060 POKE PPOS(PLR),PK-128*(PK>128)
1070 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1080 REM @ WENN STICK NICHT BEWEGT   @@
1090 REM @ BLEIBT POSITION ERHALTEN  @@
1100 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1110 IF ST=15 THEN 1000
1120 POKE PPOS(PLR),PK
1130 PPOS(PLR)=PPOS(PLR)+DIR(ST)
1140 PK=PEEK(PPOS(PLR))
1150 IF PK<>83 AND PK<>PIECE(0) AND PK<>PIECE(1)
     THEN PPOS(PLR)=PPOS(PLR)-DIR(ST
)
1160 GOTO 960
1170 POKE PPOS(PLR),PIECE(PLR)
1180 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1190 REM @@ UEBERPRUEFT OB GENAU 5    @@
1200 REM @@ STEINE IN EINER REIHE     @@
1210 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1220 FOR I=5 TO 14
1230 TEMP=PPOS(PLR):DIR=DIR(I):N=0
1240 IF DIR=0 THEN 1310
1250 TEMP=TEMP+DIR
1260 IF PEEK(TEMP)=PIECE(PLR) THEN N=N+1:GOTO 1250
1270 N=0
1280 TEMP=TEMP-DIR
1290 IF PEEK(TEMP)=PIECE(PLR) THEN N=N+1:GOTO 1280
1300 IF N=5 THEN 1360
1310 NEXT I
1320 GOTO 910
1330 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1340 REM @ UEBERPRUEFT WER GEWONNEN  @@
1350 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1360 FOR I=1 TO 100
1370 POKE 712,PEEK(53770)
1380 NEXT I
1390 POKE 712,0
1400 P=PEEK(710+PLR):C=INT(P/16)
1410 FOR I=1 TO 10
1420 POKE 710+PLR,0
1430 SOUND 0,180,9,8
1440 FOR PSE=1 TO 10:NEXT PSE
1450 POKE 710+PLR,P
1460 SOUND 0,100,10,8
1470 FOR PSE=1 TO 20:NEXT PSE
1480 NEXT I
1490 P=PEEK(711-PLR):C=INT(P/16)
1500 FOR I=14 TO 0 STEP -0.2
1510 SOUND 0,I*2,12,I
1520 POKE 711-PLR,C*16+I
1530 NEXT I
1540 POKE 711-PLR,0
1550 FOR I=1 TO 10
1560 POKE 53274,PEEK(53770)
1570 NEXT I
```

```
1580 FOR PSE=1 TO 500:NEXT PSE
1590 GRAPHICS 17:SETCOLOR 0,0,0
1600 IF PLR THEN GOTO 1680
1610 POSITION 2,1:? #6;"spieler eins ist":POSITION 6,7:? #6;"*sieger*"
1620 POSITION 6,6:? #6;"********":POSITION 6,8:? #6;"********"
1630 POSITION 0,12:? #6;"PECH SPIELER ZWEI,"
1640 POSITION 0,13:? #6;"VIELLEICHT KLAPPT"
1650 POSITION 0,14:? #6;"ES DAS NAECHSTE"
1660 POSITION 0,15:? #6;"MAL."
1670 GOTO 1740
1680 POSITION 2,1:? #6;"SPIELER ZWEI IST":POSITION 6,7:? #6;"*SIEGER*"
1690 POSITION 6,6:? #6;"@@@@@@@@":POSITION 6,8:? #6;"@@@@@@@@"
1700 POSITION 0,12:? #6;"PECH SPIELER EINS."
1710 POSITION 0,13:? #6;"VIELLEICHT KLAPPT"
1720 POSITION 0,14:? #6;"ES DAS NAECHSTE"
1730 POSITION 0,15:? #6;"MAL."
1740 FOR PSE=1 TO 300:NEXT PSE
1750 POSITION 0,22:? #6;"DRUECKE START FUER
   ":POSITION 4,23:? #6; "NEUES SPIEL"
1760 FOR PSE=1 TO 300:NEXT PSE
1770 GOSUB 380
1780 GOTO 860
1790 GOTO 1790
1800 DIM DIR(15)
1810 FOR I=5 TO 15
1820 READ A:DIR(I)=A
1830 NEXT I
1840 DATA 21,-19,1,0,19,-21,-1,0,20,-20,0
1850 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1860 REM @@  BILDSCHIRM AUFBAUEN      @@
1870 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1880 GRAPHICS 16+1:SETCOLOR 0,0,0
1890 POKE 756,226
1900 SCR=PEEK(88)+256*PEEK(89)
1910 FOR I=2 TO 20
1920 FOR J=2 TO 17
1930 POKE SCR+I*20+J,83
1940 NEXT J
1950 NEXT I
1960 FOR I=2 TO 17
1970 POKE SCR+40+I,87
1980 POKE SCR+420+I,88
1990 NEXT I
2000 FOR I=2 TO 20
2010 POKE SCR+20*I+2,65
2020 POKE SCR+20*I+17,68
2030 NEXT I
2040 POKE SCR+42,81:POKE SCR+57,69
2050 POKE SCR+422,90:POKE SCR+437,67
2060 POSITION 4,0:? #6;"FUENF GEWINNT"
2062 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
2064 REM @@ AUFBAU DES STEINCHENS      @@
2066 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
2070 DIM PPOS(1),PIECE(1),COUNT(1)
2080 PPOS(0)=SCR+229:PPOS(1)=SCR+209
2090 PIECE(0)=148:PIECE(1)=212
2100 COUNT(0)=0:COUNT(1)=0
2110 POKE PPOS(0),PIECE(0)
2120 RETURN
```

Wayne Creekmore

Mikro-Wissen griffbereit

Wer seine Mitarbeiter auf den betrieblichen Mikro-Einsatz vorbereiten wollte, hatte bisher schon verschiedene Möglichkeiten, dies zu tun. Diese Palette hat sich nun um eine neue, kostengünstige Möglichkeit erweitert: den illustrierten Leitfaden „Mikro-Wissen griffbereit" von Creekmore.
Ohne sich mit langatmigen Betrachtungen über die Entwicklung der Rechentechnik vom Abakus bis zu den EDV-Großanlagen aufzuhalten, kommt der Autor ohne Umschweife direkt zur Sache. So erfährt der Leser bereits auf der ersten Textseite, daß ein Computer nichts anderes kann als bestimmte mathematische Operationen auszuführen, daß er nicht bei allen Aufgaben optimal eingesetzt ist, daß aber niemand schneller als er die oft zitierte „Nadel im Heuhaufen" findet. Die nächste Textseite erklärt dann an einem gut gewählten Beispiel den Unterschied zwischen Hardware und Software bzw. zwischen Hardware einerseits und Betriebssystem und Anwenderprogrammen andererseits.
In der Folge werden dann Seite für Seite wichtige Vorgänge beim professionellen und kommerziellen Einsatz erläutert. So gibt die Seite 16 beispielsweise an, wie die Grundbausteine Eingabe, CPU, interner Speicher, externer Speicher und Ausgabe zusammenwirken, wenn eine editierte Datei abgespeichert werden soll und was mit der Originaldatei geschieht.
Die komprimierte, durch eine aufwendige Graphik unterstützte Darstellung hält den Leser 64 Seiten lang bei der Stange. Am Ende weiß er dann das Wichtigste über die interne Datenverarbeitung, über Computersprachen, Anwender-Software, die Hardware-Ausstattung und die Planung eines Mikro-Kaufs. Dieses Wissen gibt Sicherheit bei der Lektüre weiterführender Bücher, bei Schulungsmaßnahmen und Seminaren und beim Computer-Einsatz selbst.
Braunschweig: Vieweg/Frankfurt: Ashton-Tate 1984, DM 24,80

Datenbank

für Commodore 64

Dateiprogramme für den Commodore 64 inkl. Diskettenlaufwerk VC 1541.

Dateiprogramme gibt es ja schon eine Menge, jedoch meist nur für eine gezielte Anwendung. Das hier vorgestellte Programm gibt jedem die Möglichkeit sich sein individuelles Dateiprogramm zu erzeugen. Dabei spielt es keine Rolle, ob man Videofilme, Adressen oder etwas anderes verwalten möchte. Jeder Datensatz kann außer dem Datum der Eintragung noch bis zu 15 weitere Eintragungen enthalten. Die unterschiedlichen Kriterien werden einfach in die DATA-Zeile 60020 eingetragen. Wichtig ist, daß diese Zeile mit einem Komma abgeschlossen wird.

Z. B. Adressverwaltung:
60020 DATA Vorname, Nachname, Straße, Ort, Telefon, Geburtstag, Bemerkung.

Z. B. Videofilme:
60030 DATA Titel, Länge, Regisseur, Musik, Hauptdarsteller.

Vor dem ersten Start muß in Zeile 80 der zweite GOSUB-Befehl entfernt werden, da sonst versucht wird, eine nicht vorhandene Datei einzuladen. Nach dem Starten stehen folgende Befehle zur Verfügung:

(Z) Summieren
(O) Sortieren
(S) Suchen
(A) Daten ausgeben
(N) Neuer Antrag
(Q) Daten saven

Der Befehl „Z" summiert einen beliebigen Eintrag.
Der Befehl „O" sortiert alphabetisch nach einem bestimmten Eintrag.
Der Befehl „S" sucht ein bestimmtes Kriterium durch.
Der Befehl „A" gibt einzelne; mehrere oder auch alle Daten auf Drucker aus. Ebenfalls ist es auch möglich Adressaufkleber zu drucken.
Der Befehl „N" erlaubt einen neuen Eintrag.
Der Befehl „Q" speichert die Daten auf Diskette.
Als weitere Unterbefehle gibt es Edit zum editieren und Clear zum löschen von Einträgen.
Je nach Anzahl der Kriterien dürfte die durchschnittliche Anzahl der Datensätze bei 400 liegen. Da jedoch mit mehreren Files gearbeitet werden kann, ist diese Zahl beliebig zu vervielfachen.
Das Programm arbeitet weitgehend mit Masken und Menüs, so daß eine Fehlbedienung ausgeschlossen ist.

```
10 REM ******************
20 REM    C64 DATENBANK
30 REM ******************
50 :
60 DIM S$(16),DD$(500,16),DF$(16)
66 :
69 POKE 53281,14:POKE 53280,5:U=1
70 READ S$(U):IF S$(U)<>"" THEN U=U+1:GOTO 70
75 S$(0)="DATUM"
80 GOSUB 53000:    GOSUB 40030
100 PRINT"■■■ DATENBANK                    "
110 FORI=0 TO U-1
120 PRINT"<"I">  "TAB(9)S$(I)
130 NEXTI
135 H%=28:V%=1:GOSUB 50030:PRINTDA$
136 H%=28:V%=2:GOSUB 50030:PRINT LEFT$(F1$,10)
140 H%=0:V%=18:GOSUB 50030
145 PRINT"<Z> SUMMIEREN           <O> SORTIEREN"
150 PRINT"<S> SUCHEN              <A> DATEN AUSGEBEN"
160 PRINT"<N> NEUER EINTRAG   <Q> DATEN SAVEN"
190 PRINT:PRINT"BEFEHL -->                DATENANZ.:"O
200 GET E$:IF E$=""THEN 200
210 IF E$="N" THEN 1000
220 IF E$="Q" THEN 40140
230 IF E$="S" THEN 2000
240 IF E$="Z" THEN 3000
245 IF E$="O" THEN 4000
247 IF E$="A" THEN 6000
250 GOTO 200
1000 REM *** NEUER EINTRAG ***
1010 FI=O:DD$(FI,0)=DA$:GOSUB20000
1015 FI=O:GOSUB 30000
1017 PRINT:PRINT:PRINT" ALLES IN ORDNUNG (J/N) ?"
1018 GET E$:IF E$=""THEN 1018
1019 IF E$="N"THEN 1010
1020 O=O+1
1030 GOTO 100
2000 :
2010 REM *** SUCHEN ***
2020 :
2030 H%=0:V%=22:GOSUB 50030:PRINT"■SUCHEN■ NACH WELCHEM
     KRITERIUM";
2040 INPUT E$
2050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 2030
2060 H%=0:V%=22:GOSUB 50030:FORJ=0 TO 38:PRINT" ";:NEXTJ
2070 H%=0:V%=22:GOSUB 50030:I=VAL(E$):PRINT S$(I)" ";:M=I
2080 INPUT SW$
2090 GOSUB 51000
```

```
2095 IF FI<0 THEN 100
2100 GOSUB 20000
2110 H%=0:V%=19:GOSUB 50030:PRINT"▣ <N> NEXT <E> EDIT <M> MENUE <C> CLEAR  ▣"
2120 GET E$:IF E$="" THEN 2120
2130 IF E$="M" THEN 100
2140 IF E$="N" THEN I=M:T=T+1:GOSUB 51040:GOTO 2095
2150 IF E$="C" THEN 52000
2160 IF E$="E" THEN GOSUB 30000:DD$(FI,0)=DA$:GOTO 2095
2170 GOTO 2120
3000 :
3005 REM *** EINTRAG SUMMIEREN ***
3010 :
3030 H%=0:V%=22:GOSUB 50030:PRINT"▣ SUMME ▣ VON WELCHEM KRITERIUM
3040 INPUT E$
3050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 3030
3060 H%=0:V%=22:GOSUB 50030:FORJ=0 TO 38:PRINT" ";:NEXTJ
3061 S2=0:I=VAL(E$)
3062 FORT= 0 TO O-1:S1=VAL(DD$(T,I))
3063 S2=S2+S1:NEXTT
3070 H%=0:V%=22:GOSUB 50030:PRINT"SUMME = "S2
3080 GET E$:IFE$=""THEN3080
3090 GOTO100
4000 :
4010 REM *** SORTIEREN ****
4020 :
4030 H%=0:V%=22:GOSUB 50030:PRINT"▣SORTIEREN▣ NACH WELCHEM KRITERIUM ";
4040 INPUT E$
4050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 4030
4060 T=VAL(E$)
4070 FOR L1=0 TO O-2:FOR L2=L1+1 TO O-1
4080 GOSUB 5000
4090 IF Q1<Q2 THEN 4110
4100 GOSUB 4130
4110 NEXT:NEXT:GOTO 100
4120 REM UNTERPROGRAMME FUER BUBBLESORT
4130 FOR I=0 TO U-1
4140 DP$(I)=DD$(L1,I)
4150 DD$(L1,I)=DD$(L2,I)
4160 DD$(L2,I)=DP$(I)
4170 NEXTI:RETURN
5000 Z=1
5010 Q1$=MID$(DD$(L1,T),Z,1)
5020 Q2$=MID$(DD$(L2,T),Z,1)
5030 IF Z>15 THEN 5050
5040 IF Q1$=Q2$ THEN Z=Z+1:GOTO 5010
5050 Q1=0:IF Q1$<>"" THEN Q1=ASC(Q1$)
5060 Q2=0:IF Q2$<>"" THEN Q2=ASC(Q2$)
5070 RETURN
6000 :
6010 REM *** DATEN AUSGEBEN ***
6020 :
6030 PRINT"▣ *** DATEN AUSGEBEN ***▣"
6040 FORI=0 TO U-1
6050 PRINT"<"I">  "TAB(9)S$(I)
6060 NEXTI
6070 PRINT"< A >     ALLES   AUSGEBEN"
6080 PRINT"< E >     4 DATEN AUSGEBEN   Z.B. ADRESSE"
6090 PRINT:PRINT"BEFEHL -->";
6095 E$="":FORI=1 TO 100:NEXTI
7000 GET E$:IF E$=""THEN7000
7010 IF E$="A"THEN 8000
7020 IF E$="E"THEN 9000
7030 I=VAL(E$):
7035 OPEN 4,4
7036 PRINT#4,"LISTE VOM "DA$
```

```
7037 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
7040 FORT=0 TO O-1
7050 PRINT#4,T;CHR$(16)"05"DD$(T,I)
7060 NEXTT
7080 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
7085 CLOSE4
7090 GOTO 100
8000 :
8010 REM *** ALLES AUSGEBEN ***
8020 :
8030 OPEN 4,4
8040 PRINT#4,"GESAMTLISTE VOM "DA$
8050 PRINT#4
8060 FOR FI=0 TO O-1
8070 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
8080 FORT=0 TO U-1
8082 PRINT#4,S$(T);CHR$(16)"16: "DD$(FI,T)
8083 NEXTT
8090 NEXT FI
8095 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
8100 CLOSE4
8110 GOTO100
9000 :
9010 REM *** 4 DATEN AUSGEBEN ***
9020 :
9030 PRINT"□DREI DATEN AUSGEBEN"
9040 PRINT"________________________________________"
9041 FORI=0 TO U-1
9042 PRINT"<"I">  "TAB(9)S$(I)
9043 NEXTI
9045 PRINT"▒BEACHTEN SIE DAS DIE ERSTEN BEIDEN"
9046 PRINT"DATEN HINTEREINANDER GEDRUCKT
WERDEN !▒▒"
9050 INPUT "DATEN KENNZAHL 1 ";D1
9060 INPUT "DATEN KENNZAHL 2 ";D2
9070 INPUT "DATEN KENNZAHL 3 ";D3
9080 INPUT "DATEN KENNZAHL 4 ";D4
9100 OPEN 4,4
9110 FORT=0 TO O-1
9120 PRINT#4,DD$(T,D1);" "DD$(T,D2)
9130 PRINT#4,DD$(T,D3)
9140 PRINT#4,DD$(T,D4)
9150 PRINT#4:PRINT#4
9160 NEXTT
9170 CLOSE 4
9180 GOTO100
20000 REM *** DATENSATZ ANZEIGEN ***
20010 REM FI=NR DES DATENSATZES
20020 :
20030 PRINT"□":PRINT
20035 PRINTS$(0)": "DD$(FI,0)
20040 FORI=1 TO U-1
20050 PRINT LEFT$(S$(I),16);TAB(18)""DD$(FI,I)
20060 NEXTI:RETURN
20070 :
30000 REM *** EINGABE ROUTINE ***
30010 REM FI=NR DES DATENSATZES
30020 FORT=1 TO U-1
30030 H%=15:V%=T+1:GOSUB 50030
30040 E$="":INPUT "";E$:IF E$<>""THEN DD$(FI,T)=E$
30050 NEXTT
```

```
30070 RETURN
40010 REM *** DATEN EINLADEN ***
40020 :
40030 F2$=F1$+",S,R":OPEN 1,8,4,F2$
40040 INPUT#1,O
40050 FORI=0 TO U-1
40060 FORT=0 TO O-1
40070 INPUT#1,DD$(T,I)
40080 NEXT T,I
40090 CLOSE1
40100 RETURN
40110 :
40120 REM *** DATEN SAVEN ***
40130 :
40140 F2$="@:"+F1$+",S,W":OPEN 1,8,4,F2$
40150 PRINT"■■■ACHTUNG !!!!!!!"
40160 PRINT"■DATEN WERDEN GESPEICHERT !"
40170 PRINT#1,O
40180 FORI=0 TO U-1
40190 FORT=0 TO O-1
40195 IF DD$(T,I)="" THEN DD$(T,I)="-"
40200 PRINT#1,DD$(T,I)
40210 NEXT T,I
40220 CLOSE1:END
40230 :
50000 REM *** CURSOR SETZEN ***
50010 REM H%=HORIZONTAL
50020 REM V%=VERTIKAL
50030 PRINT"■";
50040 FORI=0 TO V%:PRINT"■";:NEXT
50045 IF H%=0 THEN 50060
50050 FORI=0 TO H%:PRINT"■";:NEXT
50060 RETURN
51000 :
51010 REM *** STICHWORT SUCHEN ***
51020 :
51030 T=0
51040 IF DD$(T,I)=SW$ THEN FI=T:RETURN
51050 T=T+1:IF T>O-1 THEN FI=-1:RETURN
51060 GOTO 51040
51070 :
52000 REM *** DATENSATZ LOESCHEN ***
52010 :
52020 O=O-1
52030 FOR T=FI TO O-1
52040 FOR I=0 TO U-1
52050 DD$(T,I)=DD$(T+1,I)
52060 NEXT I,T
52065 FOR T=0 TO U-1:DD$(O,T)="":NEXTT
52070 GOTO 100
52080 :
53000 REM *** FILENAMEN EINGEBEN ***
53010 :
53020    ? "■■ ■   D A T E N B A N K  "
53040 PRINT"■■■■■":INPUT"  FILENAMEN ";F1$
53050 PRINT"■":  INPUT"  DATUM      ";DA$
53060 RETURN
60000 :
60005 REM * IN FOLGENDE DATA-ZEILE *
60006 REM * GEWUENSCHTE ARGUMENTE  *
60007 REM * EINTRAGEN. Z.B. NAME...*
60010 :
60020 DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,
```

Tips & Tricks

Reset für die kleinen Commodore!

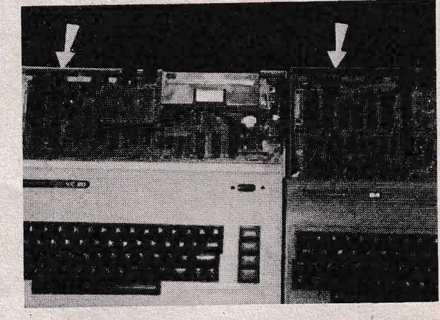
Wer kennt dies nicht?

Ich schreibe ein Programm (maschinenspr.), gebe ein paar Pok's zur Probe ein. Was ist nun los? ... Cursor futsch ... ... nichts geht mehr ...
Eine einzige Möglichkeit bleibt offen, das ist der berühmte große Knopf. – Hauptschalter aus, Hauptschalter ein, nun scheint alles wieder okay zu sein! Rein optisch laut Bildschirm ist auch alles wieder in Ordnung, aber das Programm ist und bleibt futsch!!
Ein selbstgepinntes Programm. Stunden sitze ich nun schon daran, abgespeichert war es nicht, schließlich war das Ende des Listings doch so nahe. Dann kann ich ja das ganze Programm absaven, dacht ich mir.
Aber Prosit – Mahlzeit, ich halte die Luft an, starre auf den Bildschirm voller Entsetzen, Cursor weg – nichts geht mehr, mir wirds heiß, ich spüre wie mir das Blut in den Kopf steigt, meine Gehirnzellen arbeiten plötzlich fieberhaft. Hilfe, mein Programm ist futsch! Jetzt wird mir erst bewußt, das die ganze „Tipperitis" für die Katz war. Oh', wie bin ich von meinem Computer enttäuscht. Dieser Computer, der mein bester Freund ist, entpuppt sich zum Mistkerl Nr. 1. Wie kann er mir so etwas antun?? Das schlimme an dieser Sache ist, dies war nicht das erstemal, er hat sich diese Mätzchen schon ein oder zweimal erlaubt.

Nun ist aber Schluß mit den Amokläufern

Sofort baue ich ihm eine Reset-Taste ein. Man braucht kein Elektroniker oder gar ein Computer-Mechaniker zu sein, um dies zu bewerkstelligen. Sehen wir uns hierzu die Handbücher einmal an:

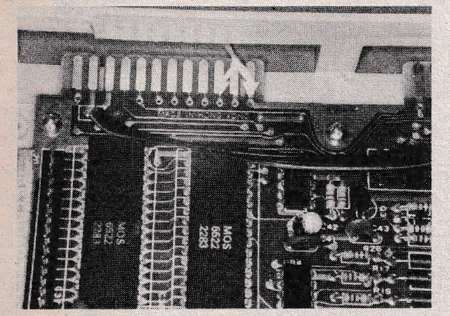
VC-20 Seite 152
CBM 64 Seite 143



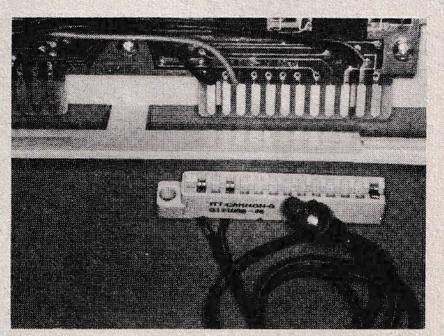
Der Commodore 64 und VC-20 geöffnet. – USER PORT – oben links (siehe Pfeil)

Dort finden wir den für uns wichtigen USER-PORT. Wir brauchen Pin 1

(Ground) und Pin 3 (Reset). Wenn wir nun die beiden Pin's kurz zusammen bringen, erreichen wir hierdurch einen „Kaltstart". Da wir den Strom


An Pin 1 + 3 (Pfeile) soll gelötet werden.

(Ein-Ausschalter) nicht abgeschaltet haben, bleibt der Speicherinhalt erhalten. Ein Basic-Programm scheint zwar verloren gegangen zu sein, kann aber durch VERBIEGEN der entsprechenden Zeiger wieder gerettet werden.
Es gibt nun zwei Möglichkeiten für den VC-20 genau wie für den 64er! RESET-TASTE innen oder außen. Nehmen wir einmal an, Sie sind außerhalb der Garantiezeit und öffnen Ihr Gerät, so sehen Sie den User-Port oben links, der Computer steht also mit der Front zu Ihnen hin gerichtet. Sie zählen die Pin's von rechts nach links, also ganz rechts ist Pin 1, links daneben Pin 2 usw. Auf der Platine sehen wir vor jedem Pin eine kleine Bohrung, dort löten wir einen dünnen isolierten etwa 50 cm langen Kabel an (Telefondraht). Das gleiche machen wir an der Bohrung vor Pin 3. Die beiden freien Drahtenden löten wir jetzt an den Taster.
Wenn wir uns jetzt einmal das Oberteil (Tastatur) von innen betrachten, stellen wir fest, daß wir hier viel Platz und mehrere Möglichkeiten haben, um den Taster zu befestigen. Mein Vorschlag: links neben der Power-Kontroll-Leuchte, aber dies bleibt Ihnen natürlich überlassen.
Sie sagen: „Ich habe noch Garantie auf meinen Rechner!" Macht knapp die Hälfte, ist meine Antwort. Sie besorgen sich einen Stecker für den User-Port und löten die beiden Drähtchen an den Stecker und den Taster. So brauchen Sie Ihr Gerät überhaupt nicht zu öffnen.

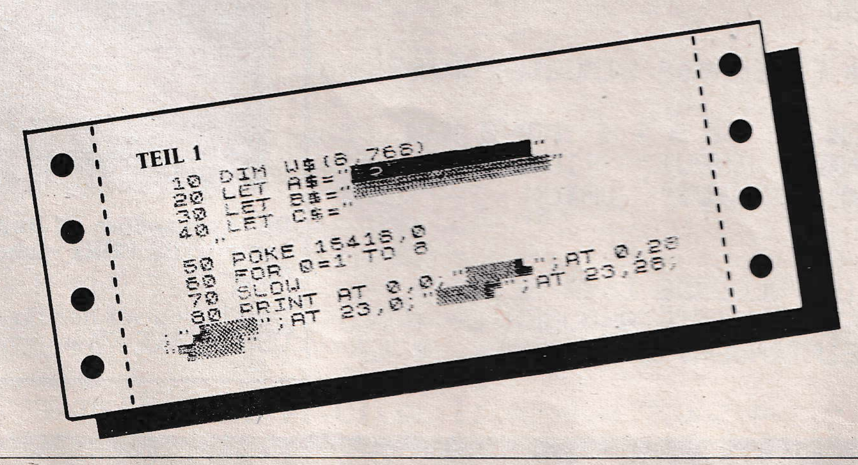

Stecker mit Kabel und Taster

Irrgarten
für ZX-81, 16 k

Am Anfang muß Teil 1 des Programmes eingegeben werden. Mit RUN wird es gestartet und auf dem Bildschirm erscheinen die möglichen 8 Bilder. Danach darf unter keinen Umständen mehr RUN benutzt werden, nur noch GOTO!!! Zur Sicherheit sollte das Programm abgespeichert werden. Nun kann das Hauptprogramm eingegeben werden. **Sinn des Spieles ist es, mit möglichst wenigen Schritten Deinen Freund aus dem Irrgarten zu leiten.** Bewahre die Übersicht und leite ihn sicher zum Ausgang! Damit es für Dich nicht zu schwer wird, kannst Du bis zu zehn mal pro Spiel eine Übersicht auf einem 20 × 20 Schritt großen Feld abrufen. Bei der Eingabe "A" läuft Dein Freund automatisch weiter, wenn es nur eine Möglichkeit gibt. Gibst Du "W" ein, wartet er auf Deine Anweisungen. Es ist manchmal ganz schön verhext, aber keine Angst, Du wirst den Ausgang schon finden!



Hier noch ein Hinweis für Computer-Fans, die einen Q-save besitzen: Das Programm erst ganz eingeben, abspeichern und dann erst laufen lassen, sonst wird die Software für den Q-save zerstört!

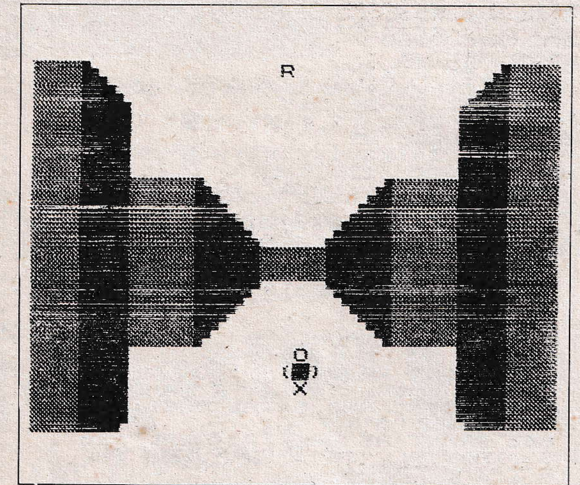

TEIL 1
```
10 DIM W$(8,768)
20 LET A$="
30 LET B$="
40 LET C$="
"
50 POKE 16418,0
60 FOR Q=1 TO 8
70 SLOW
80 PRINT AT 0,0;"    ";AT 0,28
";AT 23,0;"    ";AT 23,28;
"
```

```
  90 FOR W=1 TO 11
 100 PRINT AT W,0;"▒▒▒▒"+A$( TO W)+"▙"+C$( TO 24-2*W)+"▟"+A$( TO W)+"▒▒▒▒"
 110 PRINT AT 23-W,0;"▒▒▒▒"+A$( TO W)+"▛"+C$( TO 24-2*W)+"▜"+A$( TO W)+"▒▒▒▒"
 120 NEXT W
 130 PRINT AT 11,14;"▒▒▒▒";AT 12,14;"▒▒▒▒"
 140 IF Q=5 THEN GOTO 250
 150 FOR W=3 TO 20
 160 IF Q=1 OR Q=2 OR Q=6 THEN GOTO 180
 170 PRINT AT W,6;(C$( TO 4) AND (W<7 OR W>16))+(B$( TO 4) AND (W>6 AND W<17))
 180 IF Q=1 OR Q=3 OR Q=7 THEN GOTO 200
 190 PRINT AT W,22;(C$( TO 4) AND (W<7 OR W>16))+(B$( TO 4) AND (W>6 AND W<17))
 200 NEXT W
 210 IF Q>5 THEN GOTO 250
 220 FOR E=7 TO 16
 230 PRINT AT E,10;B$( TO 12)
 240 NEXT E
 250 FAST
 260 FOR W=0 TO 23
 270 FOR E=1 TO 32
 280 LET W$(Q,E+32*W)=CHR$ PEEK (PEEK 16396+256*PEEK 16397+E+33*W)
 290 NEXT E
 300 NEXT W
 310 NEXT Q
 320 SAVE "I"
 330 LET B$=""
 340 LET C$=""
 350 STOP
```

TEIL 2

```
  10 CLS
  20 PRINT AT VAL "5",VAL "5";"I R R G A R T E N"
  30 PRINT ,,"HELFE MIR BITTE MIT MOEGLICHST  WENIGEN SCHRITTEN AUS DIESEN              IRRGARTEN."
  40 PRINT ,,"DU KANNST ZEHNMAL EINE UEBER-   SICHT BEKOMMEN, WENN DU AUF        ""H"" DRUECKST."
  50 PRINT ,,"SOLL ICH VON SELBST WEITER       GEHEN,WENN ES NUR EINE MOEG-     LICHKEIT GIBT?"
  60 PRINT ,,"█UTOMATISCH WEITER LAUFEN."
  70 PRINT ,,"█ARTEN AUF DIE EINGABE."
  80 IF INKEY$="" THEN GOTO VAL "80"
  90 LET B$=INKEY$
 100 IF B$<>"A" AND B$<>"W" THEN GOTO VAL "80"
 110 CLS
 120 PRINT AT VAL "5",VAL "0";"BITTE ETWA 4 MINUTEN WARTEN, DERGARTEN WIRD GERADE AUFGEBAUT."
 130 PRINT ,,,,"STEUERUNG:",,,,""7"" ""5"" * ""8"",,,""6"",,,
 140 DIM I$(VAL "70",VAL "70")
 150 FOR Q=VAL "1" TO VAL "70"
 160 LET I$(VAL "1",Q)="▒"
 170 LET I$(Q,VAL "1")="▒"
 180 LET I$(Q,VAL "70")="▒"
 190 LET I$(VAL "70",Q)="▒"
 200 NEXT Q
 210 LET X=VAL "35"
 220 LET Y=X
 230 FAST
 240 FOR Q=VAL "1" TO VAL "9"
 250 IF Q=VAL "1" THEN GOTO VAL "290"
 260 LET X=INT (RND*VAL "50"+VAL "5")
 270 LET Y=INT (RND*VAL "50"+VAL "5")
 280 IF I$(X+VAL "1",Y)="█" OR I$(X-VAL "1",Y)="█" OR I$(X,Y+VAL "1")="█" OR I$(X,Y-VAL "1")="█" THEN GOTO VAL "260"
 290 LET X1=(RND<VAL ".5")*SGN (RND-VAL ".5")
 300 FOR W=VAL "1" TO INT (RND*VAL "5")+VAL "8"
 310 LET Y1=(X1<>VAL "0")*SGN (RND-VAL ".5")
 320 LET X1=(X1=VAL "0")*SGN (RND-VAL ".5")
 330 LET X1=X1*(Y1=VAL "0")
 340 LET Y1=Y1*(X1=VAL "0")
 350 FOR E=VAL "1" TO INT (RND*VAL "2")+VAL "2"
 360 LET I$(X,Y)="█"
 380 IF Y1=VAL "0" THEN IF I$(X+X1,Y-VAL "1")="█" OR I$(X+X1,Y+VAL "1")="█" THEN GOTO VAL "420"
 400 IF X1=VAL "0" THEN IF I$(X+VAL "1",Y+Y1)="█" OR I$(X-VAL "1",Y+Y1)="█" THEN GOTO VAL "420"
 410 GOTO VAL "440"
 420 IF I$(X+X1,Y+Y1)="█" THEN GOTO VAL "440"
 430 GOTO VAL "490"
 440 LET X=X+X1
 450 LET Y=Y+Y1
 460 IF Q=VAL "1" THEN IF I$(X+X1,Y+Y1)="▒" THEN GOTO VAL "530"
 470 IF I$(X,Y)="▒" THEN GOTO VAL "260"
 480 NEXT E
 490 NEXT W
 500 IF Q=VAL "1" THEN GOTO VAL "310"
 510 NEXT Q
 520 GOTO VAL "560"
 530 LET I$(X,Y)="█"
 540 LET I$(X+X1,Y+Y1)="A"
 550 GOTO VAL "510"
 570 LET X=VAL "35"
 580 LET Y=X
 590 LET N=VAL "0"
 600 LET H=VAL "10"
 610 FOR S=VAL "0" TO VAL "3"
 620 LET X1=(S=VAL "2")-(S=VAL "0")
 630 LET Y1=(S=VAL "3")-(S=VAL "1")
 640 IF I$(X+X1,Y+Y1)="█" THEN GOTO VHL "680"
 650 NEXT S
 660 LET X=X+X1
 670 LET Y=Y+Y1
 700 LET W=VAL "1"
 710 IF I$(X-Y1,Y+X1)="█" THEN LET W=W+VAL "2"
 720 IF I$(X+Y1,Y-X1)="█" THEN LET W=W+VAL "1"
 730 IF I$(X,Y)="A" THEN GOTO VAL "3000"
 740 IF I$(X+X1,Y+Y1)="A" OR I$(X+X1,Y+Y1)="█" THEN LET W=W+VAL "4"
 790 POKE VAL "16418",VAL "0"
 800 LET N=N+VAL "1"
 810 CLS
 820 SLOW
 830 PRINT AT VAL "0",VAL "0";W$(W)
 840 PRINT AT VAL "0",VAL "15";("O" AND S=VAL "0");("R" AND S=VAL "1");("U" AND S=VAL "2");("L" AND S=VAL "3")
 850 PRINT AT VAL "21",VAL "16";"O";TAB VAL "15";"(█)";TAB VAL "16";"X"
 860 FOR Q=VAL "20" TO VAL "17" STEP VAL "-1"
 870 PRINT AT Q,VAL "16";"O";TAB VAL "15";"(█)";TAB VAL "15";" X ";TAB 16;" "
 880 NEXT Q
 890 LET A$=INKEY$
 900 IF B$="A" AND A$="" THEN LET A$=("7" AND W=VAL "5")+("8" AND W=VAL "2")+("5" AND W=VAL "3")
 910 IF A$="" AND INKEY$="" THEN GOTO VAL "910"
 920 IF A$="" THEN LET A$=INKEY$
 930 IF A$="W" OR A$="A" THEN LET B$=A$
 940 LET S=S+(A$="8")-(A$="5")+VAL "2"*(A$="6")
 950 IF S<VAL "0" THEN LET S=S+VAL "4"
 960 IF S>VAL "3" THEN LET S=S-VAL "4"
 970 LET X1=(S=VAL "2")-(S=VAL "0")
 980 LET Y1=(S=VAL "1")-(S=VAL "3")
 990 IF A$="H" THEN GOTO VAL "2000"
1000 IF A$="6" THEN GOTO VAL "1300"
1010 IF E=S AND (A$="O" OR A$="P") THEN GOTO VAL "1050"
1020 IF A$="5" THEN GOTO VAL "1110"
1030 IF A$="8" THEN GOTO VAL "1200"
1040 IF A$<>"7" THEN GOTO VAL "890"
1050 IF W<VAL "5" THEN GOTO VAL "890"
1060 FOR Q=VAL "16" TO VAL "14" STEP VAL "-1"
1070 PRINT AT Q,VAL "16";"O";TAB VAL "15";"(█)";TAB VAL "15";" X ";TAB 16;" "
1080 NEXT Q
1090 PRINT AT Q+VAL "2",VAL "15";"   ";TAB VAL "16";" ";AT Q,VAL "16";".";TAB VAL "16";"A";AT Q,VAL "16";" ";TAB VAL "16";" "
1100 GOTO VAL "660"
1110 PRINT AT VAL "18",VAL "15";" █ ";TAB VAL "16";"J"
1120 IF W<>VAL "4" AND W<>VAL "3" AND W<>VAL "8" AND W<>VAL "7" THEN GOTO VAL "890"
1130 FOR Q=VAL "16" TO VAL "6" STEP VAL "-1"
1140 PRINT AT VAL "17",Q;"O ";TAB Q;"█ ";TAB Q;"J "
```

```
1150 NEXT Q
1160 PRINT AT VAL "17",VAL "6";"
 ";TAB VAL "6";" ";TAB VAL "6";"
 "
1170 GOTO VAL "650"
1200 PRINT AT VAL "18",VAL "15";
" ■ ",TAB VAL "16";"L"
1210 IF W<>VAL "2" AND W<>VAL "4
" AND W<>VAL "8" AND W<>VAL "6"
THEN GOTO VAL "890"
1220 FOR Q=VAL "16" TO VAL "24"
1230 PRINT AT VAL "17",Q;" O";TA
B Q;" ■";TAB Q;" L"
1240 NEXT Q
1250 PRINT AT VAL "17",Q;" ";TAB
 Q;" ";TAB Q;" "
1260 GOTO VAL "650"
1300 FOR Q=VAL "18" TO VAL "21"
1310 PRINT AT Q-VAL "1",VAL "15"
;"    ";AT Q,VAL "15";" O ";AT Q+
VAL "1",VAL "15";"(■)",TAB VAL "
16";"X"
```




```
1320 NEXT Q
1330 PRINT AT Q-VAL "1",VAL "16"
;" ";TAB VAL "15";"   ",TAB VAL
"16";" "
1340 GOTO VAL "660"
2000 IF H=VAL "0" THEN GOTO VAL
"890"
2010 LET H=H-VAL "1"
2020 LET EY=(Y-VAL "55")*((Y+VAL
 "15")<VAL "70")+VAL "70"
2030 LET EX=(X-VAL "60")*((X+VAL
 "10")<VAL "70")+VAL "70"
2040 LET AY=(Y-VAL "10")*((Y-VAL
 "10")>VAL "0")+VAL "1"
2050 LET AX=(X-VAL "10")*((X-VAL
 "10")>VAL "0")+VAL "1"
2060 LET I$(X,Y)="■"
2070 CLS
2080 PRINT AT VAL "2",VAL "6";
2090 FOR R=AX TO EX
2100 PRINT TAB VAL "6";I$(R,AY T
O EY)
2110 NEXT R
2120 LET I$(X,Y)="■"
2130 PRINT AT VAL "0",VAL "10";"
NOCH ";H;"MAL"
2140 IF INKEY$="" THEN GOTO VAL
"2140"
2150 LET Q=Q+VAL "1"
2160 PRINT AT VAL "0",VAL "0";W$
(W)
2170 GOTO 670
3000 PRINT AT VAL "9",VAL "9";"A
 U S G A N G"
3010 FOR Q=VAL "1" TO VAL "99"
3020 NEXT Q
3030 CLS
3040 PRINT AT VAL "5",VAL "0";"D
U MUSSTEST ";N;" SCHRITTE GEHEN"
;AT VAL "9",VAL "9";"NOCH EIN SP
IEL ?"
3050 PRINT AT VAL "15",VAL "15";
"(■",,"X"
3060 PRINT AT VAL "14",VAL "16";
"O/";AT VAL "14",VAL "17";">"
3070 IF INKEY$="J" THEN GOTO 1
3080 IF INKEY$="N" THEN STOP
3090 GOTO 3060
9000 DIM I$(1,1)
9005 SAVE "I"RRGARTEN 3D
9010 GOTO 1
```

Geldsammler

für den VC-20

Micky, ein kleines, flinkes Männchen, bekommt die Aufgabe gestellt, zwanzig Geldsäckchen einzusammeln, wofür Micky 40 Sekunden Zeit hat. Sind alle Säckchen eingesammelt, und Micky die erforderliche Punktzahl erreicht, so erscheint hierauf ein Gitter und ein neues Spielfeld. Für jedes neue Spiel haben Sie wiederum 40 Sekunden Zeit. Um das Spiel jedoch nicht zu einfach zu gestalten, wird das Sammeln von zwei Monstern behindert, die zufällig gesteuert werden. Für jedes Spiel haben Sie drei Leben, wozu Sie nach jeder geschafften Runde noch eines hinzu bekommen. Laufen Sie gegen eine Mauer bzw. gegen ein Monster, so verlieren Sie ein Leben.

Gespielt wird mit Joystick auf dem VC-20 ohne Erweiterung.

```
1 REM"■■■■■■■**************************
2 REM"■■■■■■■                    BY ANDRE KOELSCH
3 REM"■■■■■■■                    ST.WENDEL(URWEILER)
4 REM"■■■■■■■    IN DER METZ 7     ***********************
5 Q=7932:B=38652:LE=3:PU=0:POKE36878,15:J=1:HC=10:Y$="VC-20":SF=8110:GF=38830
7 GOSUB2000:POKE36879,30
10 POKE36869,255:W=7866:S=38586:R1=36874:TI$="000000":H=100
20 DATA0,0,0,0,0,0,0,0
30 DATA28,28,8,62,221,29,20,54
40 DATA48,56,16,56,56,56,16,24
50 DATA24,56,16,56,56,56,16,48
```

```
60 DATA255,145,255,165,165,255,137,255
70 DATA0,20,8,28,62,62,28,0
80 DATA129,255,129,255,129,255,129,255
90 DATA28,62,107,127,65,62,28,54
100 FORI=0TO63:READA:POKE7168+0*8+I,A:NEXT
110 PRINT"♥";:FORT=1TO130:C=INT(RND(1)*440):POKE7677+C,4:POKE38397+C,4:NEXT
120 FORT=1TO40:C=INT(RND(1)*440):POKE7677+C,5:POKE38397+C,7:NEXT
130 PRINT"■■DDDDDDDDDDDDDDDDDDDDDD";
140 FORT=1TO19:PRINT"D■■■■■■■■■■■■■■■■■■■■■D";:NEXT:TI$="000000"
150 PRINT"DDDDDDDDDDDDDDDDDDDDDDD"
160 PRINT"■■■PU."PU" ■■A■"LE" ■ "TI$
170 IFRE=20THENC=INT(RND(1)*440):RE=0:POKE7677+C,6:POKE38397+C,2:GOSUB600
180 POKEQ+X+O,0:POKEQ+X,J:POKEB+X,0
190 POKE37154,127:VB=(PEEK(37152)AND128)+(PEEK(37151)AND28)
200 IFVB=152THENJ=1:X=X-22:O=+22:Y=+22:POKE36875,230:POKE36875,0
210 IFVB=148THENJ=1:X=X+22:O=-22:Y=-22:POKE36875,230:POKE36875,0
220 IFVB=140THENJ=3:X=X-1:O=+1:Y=+1:POKE36875,200:POKE36875,0
230 IFVB=28THENJ=2:X=X+1:O=-1:Y=-1:POKE36875,200:POKE36875,0
240 M=PEEK(Q+X):IFM=5THENPU=PU+1:RE=RE+1:FORV=230TO250:POKER1+2,V:NEXT:POKER1+2,
0
250 IFM=4THENLE=LE-1:FORV=200TO220:POKER1+3,V:NEXT:POKER1+3,0:X=X+Y
253 IFTI$="000041"THEN510
255 IFM=7THENLE=LE-1:FORV=220TO240:POKER1+3,V:NEXT:POKER1+3,0:X=X+Y
260 IFM=6THENLE=LE+1:RE=0:FORT=128TO230:POKE36876,T:NEXT:POKE36876,0:GOTO110
270 IFLE<0THENGOTO510
280 POKEW+T,7:POKES+T,6:POKEW+T+Z,0
290 I%=INT(RND(1)*4)+1:ONI%GOTO292,293,294,295
292 T=T+1:Z=-1:SD=-1:GOTO298
293 T=T-1:Z=+1:SD=+1:GOTO298
294 T=T-22:Z=+22:SD=+22:GOTO298
295 T=T+22:Z=-22:SD=-22:GOTO298
298 N=PEEK(W+T):IFN=4THENT=T+SD
299 IFN=5THENT=T+SD
300 IFN=1ORN=2ORN=3THENLE=LE-1:T=T+SD:PU=PU-1:POKE36877,233:FORT=1TO50:NEXT:POKE
36877,0
301 POKESF+G,7:POKEGF+G,5:POKESF+G+P1,0
310 U%=INT(RND(1)*4)+1:ONU%GOTO312,313,314,315
311 GOTO310
312 G=G+1:P1=-1:H1=-1:GOTO316
313 G=G-1:P1=+1:H1=+1:GOTO316
314 G=G-22:P1=+22:H1=+22:GOTO316
315 G=G+22:P1=-22:H1=-22:GOTO316
316 N1=PEEK(SF+G):IFN1=4THENG=G+H1
317 IFN1=5THENG=G+H1
318 IFN1=1ORN1=2ORN1=3THENLE=LE-1:G=G+H1:PU=PU-1:POKE36877,233:FORT=1TO50:NEXT:P
OKE36877,0
500 GOTO160
510 POKE36869,240:PRINT"■■■■■■■■■■■■GAME OVER":FORT=1TO1000:NEXT:GOSUB1000
520 PRINT"■■■    ■TASTE DRUECKEN"
522 POKE198,0:WAIT198,1
530 X=0:LE=3:PU=0
540 POKE36869,255:W=7866:S=38586:GOTO110
600 :FORV=120TO250:POKER1+2,V:POKER1+2,V:NEXT:PU=PU+10:RE=0:POKER1+2,0:RETURN
1000 IFPU<HCTHENGOSUB1060
1010 IFPU>HCTHENHC=PU:GOTO1050
1011 GOTO520
1050 PRINT"■■■■IHREN NAMEN BITTE":INPUTY$
1060 PRINT"■■■■■■■■■HIGHSCORE="HC:RE=0
1070 PRINT"■■■VON "Y$:RETURN
2000 PRINT"■■■     SPIELANLEITUNG     "
2010 PRINT"■SIE MUESSEN INERHALB  ■VON 40 SECUNDEN 20    ■GOLDSAECKE EINSAMMELN"
2020 PRINT"■DANN ERSCHEINT EIN     ■GITTER DAS SIE AUCH    ■NOCH NEHMEN MUESSEN"
2030 PRINT"■HABEN SIE ES BIS DAHIN■GESCHAFT ERSCHEINT EIN■NEUES SPIELFELD"
2040 GETA$:IFA$=""THEN2040
2050 PRINT"■■■     SPIELANLEITUNG     "
2060 PRINT"■WENN SIE AUF DAS MON-  ■STER STOSSEN VERLIEREN■SIE 0-3 LEBEN"
2070 PRINT"■WENN SIE AN DIE MAUER  ■STOSSEN VERLIEREN SIE ■EIN LEBEN.BEI DREI"
2080 PRINT"■UNFAELLEN IST DAS      ■SPIEL BEENDET"
2090 GETA$:IFA$=""THEN2090
2300 PRINT"■■■     SPIELANLEITUNG     "
3000 PRINT"■GITTER    = 10 PUNKTE"
3010 PRINT"■GELDSACK = 1 PUNKT"
3015 PRINT"■BEI JEDER NEUEN RUNDE ■BEKOMMEN SIE EIN FREI-■LEBEN"
3020 GETA$:IFA$=""THEN3020
3030 RETURN
```

Zyklo

für VC-20

Ein Labyrinthspiel für die Grundversion des VC-20

Ziel ist es, in die Mitte des Labyrinths bzw. des Bildschirmes zu gelangen. Achten Sie auf dem Weg dorthin auf die sich zufällig öffnenden Türen. Sie müssen so schnell wie möglich durchlaufen werden. Schließt sich eine Tür und Sie befinden sich gerade im Durchgang, dann ist das Spiel zu Ende.
Gefahr droht Ihnen während des Spiels jedoch noch von anderer Seite. Im Labyrinth tummeln sich Monster und Gespenster. Alle machen Jagd auf Sie und versuchen mit allen Mitteln Ihnen den Weg abzuschneiden. Vorsicht! – Ein Gespenst ist dabei besonders hartnäkkig.
Versuchen Sie also, so weit wie möglich ins Labyrinth vorzudringen und den Verfolgern geschickt auszuweichen.

Eine kleine Hilfe: Im Spiel verteilt sind insgesamt vier Früchte, von denen Sie essen können. Nach dem Verzehr sind Sie kurzfristig in der Lage eine Mauer zu sprengen, wenn dies erforderlich ist.

Steuerung: links >L<, aufwärts >P<, rechts >;<, abwärts >Leertaste<. Mit >:< können Sie die Mauer sprengen.

```
1 PRINT"■■":POKE36879,10:POKE36878,15:E=7680
10 FORT=38TO150:POKE36865,T:FORTT=0TO10:NEXT:NEXT
11 PRINT"■■■■■■■■■■■SPIELANLEITUNG?(J/N)"
12 FORT=150TO38STEP-1:POKE36865,T:FORTT=0TO10:NEXT:NEXT
14 GETA$:IFA$=""GOTO14
15 IFA$="N"GOTO113
16 FORT=0TO1:FORTT=140TO240:POKE36876,TT:NEXT:NEXT:POKE36876
,0
20 PRINT"■"
22 PRINT"DAS ZIEL IST ES, ZUM":PRINT
24 PRINT"SCHATZ IN DER MITTE ":PRINT
26 PRINT"DES BILDSCHIRMS ZU GE-"
28 PRINT"LANGEN. DOCH MONSTER,":PRINT
30 PRINT"GEISTER UND TUEREN ":PRINT
32 PRINT"HINDERN SIE DARAN.":PRINT
34 FORT=0TO21:POKEE+15*22+T,83:POKE38400+15*22+T,3:POKE36877
,200:FORTT=0TO29:NEXT
36 POKE36877,0:NEXT
38 PRINT"■■■■■■■■■■■■■■-TASTE-":POKE198,0:WAIT198,1
40 PRINT"■"
42 PRINT"DIE TUEREN OEFFNEN UND"
44 PRINT"SCHLIESSEN SICH VON":PRINT
46 PRINT"ZEIT ZU ZEIT. ":PRINT
48 PRINT"■■VORSICHT! WENN SICH ":PRINT
50 PRINT"DIE TUER SCHLIESST UND"
52 PRINT"SIE BEFINDEN SICH IM ":PRINT
54 PRINT"DURCHGANG, IST DAS ":PRINT
56 PRINT"SPIEL ZU ENDE.":PRINT
58 PRINT"■■■■■■■■■■-TASTE-":POKE198,0
60 FORT=0TO21:POKEE+7*22+T,81:POKE38400+7*22+T,4:POKE36877,2
00:POKE36877,0:NEXT:WAIT198,1
62 PRINT"■"
64 FORT=150TO128STEP-4:POKE36866,T:FORTT=0TO50:NEXT:NEXT
66 PRINT"WENN SIE EINE DER ":PRINT
68 PRINT"FRUECHTE AM RAND FRES-"
70 PRINT"SEN, SO KOENNEN SIE":PRINT
72 PRINT"EINMAL DIE MAUER SPREN"
74 PRINT"-GEN.":PRINT
76 PRINT"DIES MACHEN SIE INDEM":PRINT
78 PRINT"SIE ':' UND DANN DIE":PRINT
80 PRINT"GEWUENSCHTE RICHTUNG":PRINT
82 PRINT"EINTASTEN."
83 FORT=128TO150STEP4:POKE36866,T:FORTT=0TO50:NEXT:NEXT:POKE
36866,150
84 PRINT"■■■■■■■■■■-TASTE-":POKE198,0:WAIT198,1
86 POKE36865,150:PRINT"■"
88 PRINT"■STEUERUNG:"
90 PRINT"─────────────":PRINT
92 PRINT"■■■■■■■■■■■■■■■■P"
94 PRINT"■■■■■■■■■■■|"
96 PRINT"■■■■■■■■■■L-+-;"
98 PRINT"■■■■■■■■■■■| "
99 FORT=150TO38STEP-1:POKE36865,T:FORTT=0TO10:NEXT:NEXT
100 PRINT"■■■■■■■■■SPACE"
101 PRINT"■■■■■■■■■■■■■-TASTE-":POKE198,0:WAIT198,1:PRINT"■"
102 PRINT"■■■■WENN SIE NUN EINE":PRINT
104 PRINT"TASTE DRUECKEN,WIRD ":PRINT
```

```
106 PRINT"DAS PROGRAMM GELOESCHT"
108 PRINT"UND SIE MUESSEN LOAD":PRINT
110 PRINT"EINGEBEN."
112 POKE198,0:WAIT198,1
113 PRINT"▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮BITTE ´ LOAD ´"
1000 NEW
1001 REM (C) BY JOHANN
1002 REM STRAUSS JUN.
1003 REM OSTRIACH 102
1004 REM 9570 OSSIACH
```

Teil 2

```
0 POKE36876,200
1 READA:IFA=-1GOTO20
2 POKE7168+T+8*48,A:T=T+1:GOTO1
20 FORT=0TO7:POKE7168+32*8+T,0:NEXT
30 PRINT"▮":POKE36876,0:PRINT"▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮NOCHMALS ´ LOAD
 ´":NEW
100 DATA24,126,24,60,126,90,24,60,0,127,127,127,127,127,127,
127,129,66,60,90,126,60,24
110 DATA60,24,60,90,126,126,90,165,0,255,129,129,129,129,129
,129,255,1,2,4,106,127,125
115 DATA60,24,129,126,102,90,90,102,126,129,-1
```

Teil 3

```
1 PRINT"▮▮":DIMA$(11):E=7680:Y=66:X=5:POKE36878,15:DIMO(23):
POKE36869,255:Q(0)=210:POKE36879,14
2 Q(1)=190:Q(2)=122:YY=418::FORT=0TO10:READA$(T):PRINT"□"A$(
T):NEXT:FORT=9TO0STEP-1:PRINT"□"A$(T):NEXT
15 FORT=0TO23:READO(T):POKE38400+O(T),3:NEXT:XX=10:GOSUB300
100 FORT1=0TO2:POKE36876,0
105 PE=PEEK(197):IFPE=13THENGOSUB500:GOSUB300
110 IFPE=32THENGOSUB500:GOSUB310
115 IFPE=21THENGOSUB500:GOSUB320
120 IFPE=22THENGOSUB500:GOSUB330
121 IFR=0GOTO125
122 IFPE=45THENR=1:GOSUB1000
125 A=INT(RND(1)*4)+1:POKEE+Q(T1),32:ONAGOSUB400,410,420,430
:G=G+1:IFG>8THENG=0:GOSUB590
135 W=XX:W1=YY:I=I+1:IFI>1THENI=0:GOSUB800
137 IFXX=WANDYY=W1THENI1=I1+1:GOTO139
138 I1=0:POKEE+231,54:POKE38631,4
139 IFI1>2THENGOSUB1200
140 POKEE+XX+YY,50:POKE38400+XX+YY,7:IFX=11ANDY=220THEND=1:P
=P+100:GOTO2000
245 NEXT:GOTO100
300 Y=Y-22:CI=PEEK(E+X+Y):IFCI=49ORY<0ORCI=52THENY=Y+22
302 IFCI=53THENPOKE36876,240:R=1
303 IFCI=51GOTO2000
305 POKEE+X+Y,48:POKE38400+Y+X,1:RETURN
310 Y=Y+22:CI=PEEK(E+X+Y):IFCI=49ORCI=52ORY>482THENY=Y-22
312 IFCI=53THENPOKE36876,240:R=1
313 IFCI=51GOTO2000
315 POKEE+X+Y,48:POKE38400+X+Y,1:RETURN
320 X=X-1:CI=PEEK(E+X+Y):IFCI=49ORCI=52ORX<0THENX=X+1
322 IFCI=53THENPOKE36876,240:R=1
323 IFCI=51GOTO2000
325 POKEE+Y+X,48:POKE38400+Y+X,1:RETURN
330 X=X+1:CI=PEEK(E+Y+X):IFCI=49ORCI=52ORX>21THENX=X-1
332 IFCI=53THENPOKE36876,240:R=1
333 IFCI=51GOTO2000
335 POKEE+Y+X,48:POKE38400+Y+X,1:RETURN
400 Q(T1)=Q(T1)-22:IFPEEK(E+Q(T1))=48GOTO2000
405 IFPEEK(E+Q(T1))<>32THENQ(T1)=Q(T1)+22
409 POKEE+Q(T1),51:POKE38400+Q(T1),5:RETURN
410 Q(T1)=Q(T1)+22:IFPEEK(E+Q(T1))=48GOTO2000
415 IFPEEK(E+Q(T1))<>32THENQ(T1)=Q(T1)-22
419 POKEE+Q(T1),51:POKE38400+Q(T1),5:RETURN
420 Q(T1)=Q(T1)-1:IFPEEK(E+Q(T1))=48GOTO2000
425 IFPEEK(E+Q(T1))<>32THENQ(T1)=Q(T1)+1
429 POKEE+Q(T1),51:POKE38400+Q(T1),5:RETURN
430 Q(T1)=Q(T1)+1:IFPEEK(E+Q(T1))=48GOTO2000
435 IFPEEK(E+Q(T1))<>32THENQ(T1)=Q(T1)-1
439 POKEE+Q(T1),51:POKE38400+Q(T1),5:RETURN
500 POKEE+X+Y,32:POKE36876,200:RETURN
```

```
590  IFPEEK(E+O(O1))=48ORPEEK(E+O(O2))=48GOTO2000
600  POKEE+O(O1),52:POKEO(O2)+E,52:POKE36876,180:O1=INT(RND(1
)*23):O2=INT(RND(1)*24)
605  POKEE+O(O1),32:POKEE+O(O2),32:P=P-2:RETURN
800  POKEE+XX+YY,32:IFYY<YTHENYY=YY+22:GOSUB900:IFJTHENYY=YY-
22
805  IFYY>YTHENYY=YY-22:GOSUB900:IFJTHENYY=YY+22
810  IFXX>XTHENXX=XX-1:GOSUB900:IFJTHENXX=XX+1
815  IFXX<XTHENXX=XX+1:GOSUB900:IFJTHENXX=XX-1
817  RETURN
900  IFPEEK(E+XX+YY)=48GOTO2000
905  IFPEEK(E+XX+YY)<>32THENJ=1:RETURN
910  J=0:RETURN
1000 POKE36876,230:GOSUB1100:P=P+10:POKEE+Y+X+G,32:R=0:RETUR
N
1100 SS=PEEK(197):IFSS=32THENG=+22:RETURN
1105 IFSS=13THENG=-22:RETURN
1110 IFSS=21THENG=-1:RETURN
1115 IFSS=22THENG=1:RETURN
1120 GOTO1100
1200 U=INT(RND(1)*4)-1:IFU=0THENU=22:GOTO1400
1205 IFU=2THENU=-22:GOTO1400
1300 XX=XX+U:CI=PEEK(E+XX+YY):IFCI=49ORCI=52ORXX>21ORXX<0THE
NXX=XX-U:RETURN
1305 POKEE+XX+YY,50:POKEE+XX-U+YY,32:GOTO1300
1400 YY=YY+U:CI=PEEK(E+XX+YY):IFCI=49ORCI=52ORYY>484ORYY<0TH
ENYY=YY-U:RETURN
1405 POKEE+XX+YY,50:POKEE+XX+YY-U,32:RETURN
2000 FORT=240TO140STEP-1:POKE36876,T:NEXT:POKE36869,240:PRIN
T"■":POKE36877,0
2005 PRINT"■■■■DAS SPIEL IST AUS":IFD=0GOTO2020
2010 FORT=140TO240:POKE36876,T:NEXT:PRINT"■■BRAVO!!":GOTO203
0
2020 PRINT"■■VERSAGER!!"
2030 PRINT"■■PUNKTE:"P+100:POKE36876,0:POKE198,0:PRINT"■■■■■
NOCH EINMAL?"
2035 GETA$:IFA$=""GOTO2035
2040 IFA$="J"THENRUN
2050 END
60000 DATA"5  111111111111111111 5","    1                    1  "
60005 DATA"   4  111111111111  4  ","    1  4             4  1  ",
"   1 11 1114111 11 1  "
60010 DATA"   1 1              1 1  ","    1   114111411   1  ",
"   1 1 1        1 1 1  "
60015 DATA"   1 1 4 11411 4 1 1  ","    1 1 1 1   1 1 1 1  ",
"   1 4 1 4 6 4 1 4 1  "
60020 DATA47,63,72,82,99,141,145,183,187,191,225,229,233,237
,271,275,279,317,321,363
60025 DATA380,390,399,415
```


Impressum

Homecomputer
erscheint monatlich im
Tronic-Verlag, 3444 Wehretal 1

Redaktion:
Axel Gredé (verantwortlich),
Rainer Beck, Frank Brall

Freie Mitarbeiter:
Volker Becker, Holger Gredé

Gesamtherstellung:
D+V Druckhaus Dierichs Kassel

Vertrieb:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden
Telefon 0 61 21 / 26 60

Anfragen nicht an Vertrieb oder Druckerei, sondern nur an den Verlag!

Anzeigenleitung:
Annelie Kratzenberg

Erscheinungsweise:
Erstverkaufstag von HOMECOMPUTER ist Anfang des Monats.

Anzeigenpreise:
Bitte Mediaunterlagen anfordern.

Anzeigenannahmeschluß:
Jeweils 3 Wochen vor Erscheinungstermin.

Urheberrecht:
Alle in HOMECOMPUTER veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten. Reproduktionen jeder Art (Fotokopien, Microfilm, Erfassung in Datenverarbeitungsanlagen usw.) bedürfen der schriftlichen Genehmigung des Verlags. Alle veröffentlichte Software wurde von Mitarbeitern des Verlages oder von freien Mitarbeitern erstellt.
Aus ihrer Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder Bezeichnungen frei von Schutzrechten sind.

Bezugspreise:
Einzelheft 5,50 DM
Abonnement: Inland 55,– DM im Jahr
(12 Ausgaben)
Ausland: Europa 80,– DM, USA 110,– DM

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.
Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von 100,– DM pro abgedruckter Seite im Heft einverstanden sind.
Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.
Rücksendung erfolgt nur gegen Erstattung der Unkosten. Zusendungen von Software zur Veröffentlichung sollten folgendes enthalten:
Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf und ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaus, Farbe, Grafik usw.). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

IVW geprüft.

PLONG!

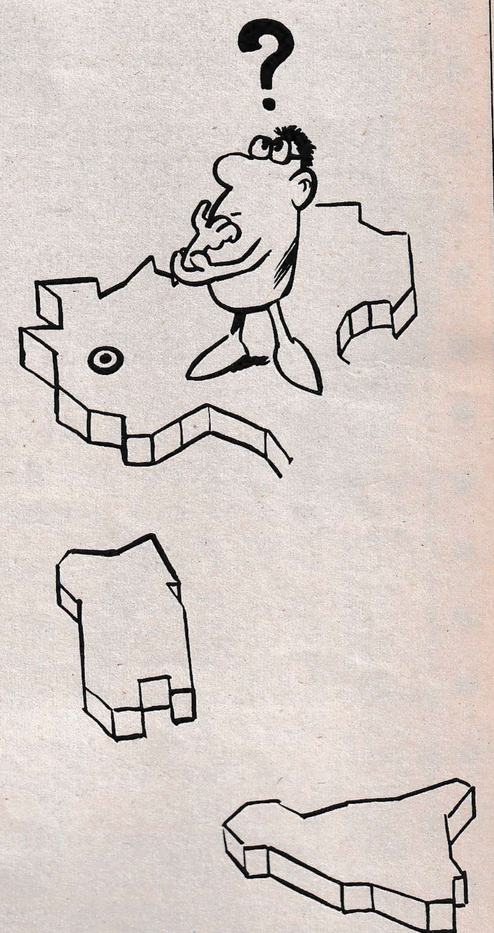
für ZX-Spectrum

Bei diesem Programm handelt es sich um ein Abenteuerspiel, in dem die Spielfigur vier Etagen überwinden muß, um den nächsten Level zu erreichen

Zur Programmsteuerung
Nachdem Sie das Programm mit „RUN“ gestartet haben, erscheint zunächst das Titelbild in Laufschrift mit einer kurzen Anleitung.
Drücken Sie nun eine beliebige Taste, so wird der Bildschirm gelöscht, das Spielfeld des Programmes aufgebaut und ihre Spielfigur in der rechten unteren Ecke des Bildschirms dargestellt. Ihre Figur können Sie nun mit Hilfe der Tasten „O“ und „P“ nach rechts bzw. links steuern. Wird zusätzlich zu diesen beiden Tasten „CAPS-SHIFT“ gedrückt, so springt die Figur in die entsprechende Richtung.

Der Bildschirmaufbau
Die untere Etage stellt einen Sumpf dar, der aus Wellen und Büschen besteht und bei dem nur noch an den Seiten fester Boden vorhanden ist. Sie müssen nun versuchen die Spielfigur durch den Sumpf zu bringen, ohne hierbei über eine Welle zu gehen, da Sie nur auf den Büschen des Sumpfes stehen können. Zu allem Unglück verschwinden diese jedoch ab und zu, so daß immer einer der Büsche fehlt. Es ist somit auch nicht sicher, daß Sie die ganze Zeit über auf einem Busch stehen bleiben können. Zu all dem kommt noch hinzu, daß in dieser Etage plötzlich auftauchende Geldsäcke einzusammeln sind. Haben Sie mindestens acht dieser Geldsäcke eingesammelt und stehen auf dem roten Malzeichen, so kommen Sie in die ZWEITE ETAGE.
In dieser Etage sollen Sie wiederum mindestens einen der bekannten Geldsäcke einsammeln, bevor Sie durch die Berührung des roten Malzeichens in die dritte Etage kommen. Aber Vorsicht, denn in dieser Ebene bewegen sich sechs Feuerbälle von links nach rechts, die Sie überspringen müssen um nicht zu verbrennen. In der DRITTEN ETAGE gilt es gefährliche Nagelvorhänge zu passieren, die von der Decke bis zum Boden hinabreichen und die ständig durch eine von oben nach unten laufende Lücke unterbrochen werden. Sie müssen nun abwarten, bis diese Lücke sich in einer Höhe befindet, in der Sie hindurchspringen können. Haben Sie alle Vorhänge passiert, so kommen Sie wiederum durch berühren des roten Malzeichens in die VIERTE ETAGE.
In dieser Etage, die gleichzeitig auch die letzte ist, haben Sie die Aufgabe, Löcher, die zufällig gelegt werden, zu stopfen. Hierzu müssen Sie über den blauen Stein, der links im Bild abgebildet ist, gehen, um einen Stein aufzunehmen. Haben Sie einen Stein aufgenommen, so müssen Sie ihn nun zu einem Loch bringen, um es zu stopfen. Passen Sie jedoch auf, denn Sie können nur über bereits gestopfte Löcher gehen. Da Sie die Steine nur locker in die Löcher legen, ist es ratsam, nach einem Sprung, nicht auf diese Steine zu springen, da diese dann durch die Löcher hindurchfallen würden und Sie ein weiteres Leben verloren hätten. Um Ihnen diese Etage nicht zu leicht zu machen, rollt ständig ein Ball von rechts nach links, der alles was ihm den Weg versperrt, überrollt. Berühren Sie nun das rote Malzeichen, so haben Sie den ersten Level gemeistert.

Punktewertung
Jeder eingesammelte Geldsack bringt Ihnen 50 Punkte. Zusätzlich gibt es noch verschiedene Bonuswerte. Nach jeder gemeisterten Etage wird am unteren Bildschirmrand ein Symbol (Totenschädel, Herz, Pik) angezeigt, das für jeweils einen anderen Bonuswert steht (20, 130, 80) der Ihrem Punktekonto gutgeschrieben wird. Haben Sie jedoch in einem Level drei Totenköpfe als Bonus erhalten, wo wird ihr Konto um 200 Punkte vermindert.
Das Spiel ist zu Ende, wenn Sie alle fünf Leben verwirkt haben oder die 1200 Zeiteinheiten überschritten sind.



Zur Eingabe in den Computer
Besitzen Sie einen ZX-Spectrum mit 16 k, so sollten Sie vor der Eingabe, bzw. dem Laden des Programms kurz den Netzstecker herausziehen, um sicherzugehen, daß der Speicher völlig leer ist. Generell jedoch müssen die Grafikzeichen der Zeilen 110–125 in verschiedenen Farben eingegeben werden. Die Umschaltung hierzu erreichen Sie durch drücken der „CAPS-SHIFT“-Taste mit der entsprechenden Farbtaste im erweiterten Modus.
Und nun ... viel Spaß.

```
   10 GO SUB 5000: BORDER 0: PAPE
R 0: BRIGHT 1: INK 7: CLS
   11 GO SUB 6100
   14 DIM c(3): DIM b(12): DIM p(
4): DIM d(7)
   20 GO SUB 100
   30 GO TO 200
  100 CLS
  110 PRINT AT 21,0;"ADACBCBCBCBC
BCBCBCBCBCBCBCBCAAAA"
  115 PRINT AT 16,0;"AAAAAAAAAAAA
AAAAAAAAAAAAAAAAAADA"
  120 PRINT AT 10,0;"ADAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA"
  125 PRINT AT 4,0;"AAAAAAAAAAAAA
AAAAAAAAAAAAAAAAADA"
  130 FOR n=0 TO 21: PRINT AT n,0
; INK 1;"K";AT n,31;"K": NEXT n
  135 PRINT AT 0,2;"PUNKTE: 0";AT
 0,20;"ZEIT: 0"
  136 PRINT #0;AT 0,4; INK 4;"EEE
E"
  190 LET c=0: FOR n=4 TO 26 STEP
 2
  195 LET c=c+1: LET b(c)=n: NEXT
 n
  196 RETURN
  200 LET ll=1: LET ti=0: LET q1=
10: LET q2=5: LET q3=.25
  201 LET bo=0: LET zt=0: LET ab=
0: LET le=5: LET a1=4: LET u=1:
LET t=0: LET a=1
  202 LET f=0: LET c2=0: LET b$="
JLM": LET c$="   I   I   I   I
  I   I"
  203 LET x=30: LET y=20: LET x1=
x: LET y1=y: LET a$="F"
  220 IF INKEY$="P" THEN LET a$="
E": LET m=1: GO TO 500
  225 IF INKEY$="O" THEN LET a$="
F": LET m=-1: GO TO 500
  230 IF INKEY$="p" THEN LET x=x+
1: LET a$="E"
  235 IF INKEY$="o" THEN LET x=x-
1: LET a$="F"
  240 GO SUB 500+30*(y>16)+200*(y
>10 AND y<17)+400*(y>4 AND y<11)
+600*(y<4)
  250 PRINT AT 0,10;p(1)+p(2)+p(3
)+p(4)
  260 PRINT AT 0,26;INT ti
  265 LET ti=ti+.2
  270 IF ti>=1200 THEN GO SUB 300
0
  395 IF x>30 OR x<1 THEN LET x=x
1
  400 PRINT AT y,x;a$
  405 IF x=x1 AND y=y1 THEN GO TO
 420
  410 PRINT AT y1,x1;" "
  420 LET x1=x: LET y1=y
  430 GO TO 220
  500 IF x+2*m>30 OR x+2*m<1 THEN
 GO TO 236
  505 PRINT AT y,x;" ": BEEP .002
,40: GO SUB 521: PRINT AT y-1,x+
m;a$: BEEP .003,45: PRINT AT y-1
,x+m;" ": LET x=x+2*m: GO SUB 78
0: GO SUB 6000: BEEP .004,50: PR
INT AT y,x;a$
  520 GO TO 236
  521 IF ATTR (y-1,x+m)=66 THEN B
EEP .07,55: BEEP .03,58: BEEP .0
5,53: LET zt=zt+1: LET p(u)=p(u)
+50
  522 IF ATTR (y-1,x+m)=70 THEN F
OR n=5 TO 9: PRINT AT n,x4;" ":
NEXT n: LET x4=27: GO SUB 650
  528 RETURN
  530 IF ATTR (21,x)=69 THEN GO S
UB 650
  535 LET t=t+1
  536 IF t=q1 THEN LET t=0: GO TO
 540
  537 GO TO 560
  540 PRINT AT 21,b(a); INK 5; PA
PER 1;"B"; PAPER 0; INK 7;AT 19,
b(a1)-1;" "
  545 LET a=1+INT (RND*12)
  550 PRINT AT 21,b(a); INK 5;"C"
  555 LET a1=1+INT (RND*12): PRIN
T AT 19,b(a1)-1; INK 2;"G"
  560 IF ATTR (21,x1)=66 AND zt>=
8 THEN LET zt=0: PRINT AT 19,b(a
1)-1;" ": LET t=0: LET a$="E": G
O TO 3200
  640 RETURN
  641 LET f=0: FOR n=1 TO 7: PRIN
T AT 4,d(n);" ": NEXT n
  650 PRINT AT y1,x1;" "
  651 FOR n=1 TO 9
  655 POKE 23606,0: PRINT AT y,x;
"x"
  656 BEEP .03,-11
  660 POKE 23606,1: PRINT AT y,x;
"x"
  661 BEEP .02,-10
  670 NEXT n
  671 LET le=le-1
  672 PRINT #0,AT 0,3+le;" "
  673 POKE 23606,0
  674 IF le=0 THEN GO TO 3000
  675 PRINT AT y,x;" ": IF ATTR (
y+1,1)=66 THEN LET x=30: LET a$=
"F": RETURN
  680 LET x=1: LET a$="E": RETURN
  700 GO SUB 780
  701 LET c2=c2+1
  702 IF c2=q2 THEN LET c2=0: GO
TO 704
  703 GO TO 720
  704 LET t=t+1
  705 PRINT AT 15,4; INK 4;c$(t T
O LEN c$);c$( TO t-1)
  706 PRINT AT 14,a;" "
  707 LET a=4+INT (RND*26): PRINT
 AT 14,a; INK 2;"G"
  710 IF t=LEN c$ THEN LET t=0
  720 GO SUB 780
  730 IF ATTR (16,x1)=66 AND zt>=
1 THEN PRINT AT 14,a;" ": LET t=
0: LET x4=27: LET a$="F": FOR n=
5 TO 9: PRINT AT n,x4; INK 6;"N"
: NEXT n: GO TO 3200
  770 RETURN
  780 IF x<1 OR x>30 THEN RETURN
  782 IF ATTR (y,x)<>71 AND POINT
 (x*8,49)=1 THEN GO TO 650
  785 RETURN
  900 IF x4<6 THEN GO TO 940
  905 PRINT AT 5+t,x4; INK 6;"N";
AT 6+t,x4;" "
  910 PRINT AT 5+t,x4; INK 6;"N";
AT 6+t,x4; INK 7;" "
  920 LET t=t+1
  930 IF t=4 THEN PRINT AT 9,x4;
INK 6;"N": LET t=0
  935 IF x<x4 THEN FOR n=5 TO 9:
PRINT AT n,x4;" ": NEXT n: LET x
4=x4-(3+INT (RND*3)): IF x4>=6 T
HEN FOR n=5 TO 9: PRINT AT n,x4;
 INK 6;"N": NEXT n
  940 GO SUB 6000
  950 IF ATTR (10,x1)=66 THEN LET
 t=0: LET a$="E": GO TO 3200
  990 RETURN
 1100 IF ab=0 AND ATTR (4,x)=71 T
HEN GO SUB 641
 1110 IF ab=1 AND ATTR (4,x)=71 T
HEN LET f=f+1: LET ab=0: LET u=u
+20: BEEP .03,-10: BEEP .01,-20:
 PRINT AT 4,x; INK 5; PAPER 2;"A
"
 1120 IF ATTR (4,x)=69 THEN LET a
b=1: BEEP .03,-12: BEEP .006,30:
 BEEP .01,-3
 1145 LET t=t+q3
 1150 PRINT AT 3,29-t; INK 5;"H";
 INK 7;" "
 1160 IF 29-t=1 THEN PRINT AT 3,2
9-t;" ": LET t=0
 1170 IF INT (29-t)=x THEN GO SUB
 641
 1180 IF ATTR (4,x1)=66 AND f=7 T
HEN GO TO 4000
 1300 RETURN
 3000 PRINT AT 12,11; INK 4;"DAS
WAR'S !"; INK 2;AT 14,9;"TASTE 1
 ODER 2"
 3010 FOR n=1 TO 3
 3020 LET p(1)=p(1)+20*(c(n)=1)+1
30*(c(n)=2)+80*(c(n)=3): NEXT n
 3030 IF c(1)=1 AND c(2)=1 AND c(
3)=1 THEN LET p(1)=p(1)-200
 3040 PRINT AT 0,10;p(1)+p(2)+p(3
)+p(4): PRINT #0;AT 0,10;"
 "
 3045 PRINT AT 1,0;"Leistungsquot
ient: ";(p(1)+p(2)+p(3)+p(4))*ll
/INT ti
 3050 PRINT #0;AT 0,1; INK 5;"O"
 3055 IF INKEY$="1" THEN RUN 11
 3056 IF INKEY$="2" THEN RUN 14
 3060 PRINT #0;AT 0,1;" "
 3070 GO TO 3050
 3200 FOR n=-10 TO 20: BEEP .01,n
: BEEP .003,20: NEXT n
 3210 LET u=u+1: LET a=16: LET bo
=bo+2: LET c=1+INT (RND*3): PRIN
T #0;AT 0,8+bo; INK c;b$(c): LET
 c(bo/2)=c
```

Weiter Seite 73

Grand-Prix


für ZX-81

Ein Spiel für den ZX-81 in Maschinensprache geschrieben

Grand-Prix ist das allseits bekannte Automobilrennen. In einer Ihnen vorgeschriebenen Zeit müssen Sie eine bestimmte Wegstrecke zurücklegen. Aber Vorsicht! Es kommen Ihnen Fahrzeuge entgegen, denen Sie ausweichen müssen. Haben Sie einige Runden geschafft und sind mit keinem Auto kollidiert, kommen Sie in die Punktwertung. Ihren Rennwagen steuern Sie mit den Tasten 5 und 8.

Wir glauben, daß einigen ZX-81-Fans Grand-Prix immer noch in der Sammlung fehlt. Darum hoffen wir, liegen wir richtig, das Programm in unser Heft aufzunehmen.

Wir jedenfalls wünschen eine gute Fahrt!

```
FUER DIE MASCHINEN-SPRACHE
MUESSEN SIE 545 ZEICHEN RESERVIEREN.
AM BESTEN IN DER ERSTEN ZEILE
  1 REM  (545 ZEICHEN)
DANN BENUTZEN SIE FOLGENDE
MC-LADE-ROUTINE :
 10 FOR X=16514 TO 17058
 20 SCROLL
 30 INPUT Y
 40 SCROLL
 50 PRINT X,Y : Poke X,Y
 60 NEXT X
WENN SIE ALLES EINGEGEBEN HABEN
DANN LOESCHEN SIE DIE ZEILEN
VON 10 BIS 60
UND SAVEN DAS MC-PROGRAMM
DANACH GEBEN SIE DIE RESTLICHEN
PROGRAMMZEILEN AB ZEILE 80 EIN.
```

```
  90 LET HS=0
 100 POKE 16418,0
 110 PRINT AT 22,0;"SCORE:0    ";TAB 1
     6;"HIGH SCORE:";HS;"0";AT 0,0;
 120 RAND USR 16989
 130 LET H=(PEEK 16446+256*PEEK 16447)
 140 IF H>HS THEN LET HS=H
 150 IF INKEY$<>"" THEN GOTO 150
 160 IF INKEY$="" THEN GOTO 160
 180 GOTO 110
9000 FOR F=16514 TO 17058 STEP 5
9010 PRINT F;
9020 LET A=7
9030 FOR G=F TO F+4
9040 PRINT TAB A;PEEK G;
9050 LET A=A+5
9060 NEXT G
```

```
9070 PRINT
9080 NEXT F
9090 STOP
9100 SAVE "GRAND-PRIX"
9110 RUN
```

```
16514 - 128
16515 - 128
16516 - 128
16517 - 5
16518 - 150
16519 - 133
16520 - 128
16521 - 38
16522 - 128
16523 - 5
16524 - 148
16525 - 133
16526 - 128
16527 - 128
16528 - 128
16529 - 5
16530 - 148
16531 - 133
16532 - 128
16533 - 59
16534 - 128
16535 - 5
16536 - 150
16537 - 133
16538 - 128
16539 - 128
16540 - 128
16541 - 135
16542 - 131
16543 - 4
16544 - 133
16545 - 136
16546 - 5
16547 - 2
16548 - 3
16549 - 1
16550 - 14
16551 - 22
16552 - 6
16553 - 10
16554 - 62
16555 - 8
16556 - 215
16557 - 16
16558 - 253
16559 - 6
16560 - 12
16561 - 62
16562 - 128
16563 - 215
16564 - 16
16565 - 253
16566 - 6
16567 - 10
16568 - 62
16569 - 8
16570 - 215
16571 - 16
16572 - 253
16573 - 13
16574 - 32
16575 - 232
16576 - 33
16577 - 14
16578 - 12
16579 - 34
16580 - 60
16581 - 64
16582 - 33
16583 - 0
16584 - 0
16585 - 34
16586 - 62
16587 - 64
16588 - 33
16589 - 66
16590 - 64
16591 - 6
16592 - 16
16593 - 54
16594 - 0
16595 - 35
16596 - 16
16597 - 251
16598 - 62
16599 - 1
16600 - 50
16601 - 64
16602 - 64
16603 - 62
16604 - 255
16605 - 237
16606 - 91
16607 - 60
16608 - 64
16609 - 205
16610 - 133
16611 - 66
16612 - 17
16613 - 130
16614 - 64
16615 - 205
16616 - 31
16617 - 66
16618 - 237
16619 - 91
16620 - 60
16621 - 64
16622 - 205
16623 - 133
16624 - 66
16625 - 6
16626 - 2
16627 - 126
```

```
16628 - 254
16629 - 128
16630 - 40
16631 - 9
16632 - 254
16633 - 5
16634 - 40
16635 - 5
16636 - 254
16637 - 133
16638 - 54
16639 - 0
16640 - 192
16641 - 35
16642 - 35
16643 - 126
16644 - 254
16645 - 128
16646 - 40
16647 - 9
16648 - 254
16649 - 133
16650 - 40
16651 - 3
16652 - 254
16653 - 5
16654 - 54
16655 - 0
16656 - 192
16657 - 17
16658 - 64
16659 - 0
16660 - 25
16661 - 16
16662 - 220
16663 - 17
16664 - 98
16665 - 0
16666 - 237
16667 - 82
16668 - 62
16669 - 38
16670 - 190
16671 - 192
16672 - 62
16673 - 40
16674 - 6
16675 - 50
16676 - 16
16677 - 254
16678 - 61
16679 - 32
16680 - 249
16681 - 42
16682 - 12
16683 - 64
16684 - 17
16685 - 16
16686 - 0
16687 - 25
16688 - 17
16689 - 33
16690 - 0
16691 - 58
16692 - 62
16693 - 64
16694 - 15
16695 - 48
16696 - 3
16697 - 54
16698 - 5
16699 - 25
16700 - 6
16701 - 11
16702 - 54
16703 - 128
16704 - 25
16705 - 54
16706 - 5
16707 - 25
16708 - 16
16709 - 248
16710 - 205
16711 - 187
16712 - 2
16713 - 237
16714 - 91
16715 - 60
16716 - 64
16717 - 124
16718 - 254
16719 - 247
16720 - 32
16721 - 20
16722 - 205
16723 - 133
16724 - 66
16725 - 1
16726 - 66
16727 - 0
16728 - 54
16729 - 128
16730 - 9
16731 - 54
16732 - 128
16733 - 28
16734 - 123
16735 - 254
16736 - 20
16737 - 32
16738 - 1
16739 - 29
16740 - 24
16741 - 25
16742 - 124
16743 - 254
16744 - 223
16745 - 32
16746 - 20
16747 - 205
16748 - 133
16749 - 66
16750 - 35
16751 - 35
16752 - 1
16753 - 66
16754 - 0
16755 - 54
16756 - 128
16757 - 9
```

```
16758 - 54
16759 - 128
16760 - 29
16761 - 123
16762 - 254
16763 - 9
16764 - 32
16765 - 1
16766 - 28
16767 - 237
16768 - 83
16769 - 60
16770 - 64
16771 - 205
16772 - 133
16773 - 66
16774 - 62
16775 - 255
16776 - 17
16777 - 130
16778 - 64
16779 - 205
16780 - 31
16781 - 66
16782 - 1
16783 - 6
16784 - 22
16785 - 205
16786 - 245
16787 - 8
16788 - 237
16789 - 75
16790 - 62
16791 - 64
16792 - 3
16793 - 237
16794 - 67
16795 - 62
16796 - 64
16797 - 205
16798 - 152
16799 - 10
16800 - 62
16801 - 28
16802 - 215
16803 - 58
16804 - 62
16805 - 64
16806 - 203
16807 - 191
16808 - 254
16809 - 127
16810 - 58
16811 - 64
16812 - 64
16813 - 32
16814 - 9
16815 - 60
16816 - 254
16817 - 5
16818 - 32
16819 - 1
16820 - 61
16821 - 50
16822 - 64
16823 - 64
16824 - 71
16825 - 33
16826 - 66
16827 - 64
16828 - 197
16829 - 203
16830 - 70
16831 - 32
16832 - 36
16833 - 203
16834 - 198
16835 - 58
16836 - 52
16837 - 64
16838 - 15
16839 - 48
16840 - 4
16841 - 203
16842 - 206
16843 - 24
16844 - 2
16845 - 203
16846 - 142
16847 - 230
16848 - 3
16849 - 30
16850 - 7
16851 - 60
16852 - 71
16853 - 62
16854 - 0
16855 - 198
16856 - 3
16857 - 16
16858 - 252
16859 - 131
16860 - 95
16861 - 22
16862 - 253
16863 - 35
16864 - 115
16865 - 35
16866 - 114
16867 - 43
16868 - 43
16869 - 203
16870 - 78
16871 - 32
16872 - 10
16873 - 58
16874 - 62
16875 - 64
16876 - 15
16877 - 48
16878 - 4
16879 - 35
16880 - 35
16881 - 24
16882 - 37
16883 - 35
16884 - 94
16885 - 35
16886 - 86
16887 - 229
```

```
16888 - 213
16889 - 122
16890 - 254
16891 - 23
16892 - 56
16893 - 2
16894 - 22
16895 - 0
16896 - 205
16897 - 133
16898 - 66
16899 - 17
16900 - 142
16901 - 64
16902 - 205
16903 - 31
16904 - 66
16905 - 209
16906 - 225
16907 - 20
16908 - 122
16909 - 114
16910 - 254
16911 - 22
16912 - 32
16913 - 6
16914 - 43
16915 - 43
16916 - 203
16917 - 134
16918 - 35
16919 - 35
16920 - 35
16921 - 193
16922 - 16
16923 - 160
16924 - 195
16925 - 234
16926 - 64
16927 - 235
16928 - 6
16929 - 4
16930 - 254
16931 - 253
16932 - 32
16933 - 5
16934 - 5
16935 - 35
16936 - 35
16937 - 35
16938 - 60
16939 - 254
16940 - 254
16941 - 32
16942 - 5
16943 - 35
16944 - 35
16945 - 35
16946 - 5
16947 - 60
16948 - 254
16949 - 255
16950 - 32
16951 - 4
16952 - 35
16953 - 35
16954 - 35
16955 - 5
16956 - 254
16957 - 19
16958 - 32
16959 - 1
16960 - 5
16961 - 254
16962 - 20
16963 - 32
16964 - 2
16965 - 5
16966 - 5
16967 - 254
16968 - 21
16969 - 32
16970 - 2
16971 - 6
16972 - 1
16973 - 197
16974 - 1
16975 - 3
16976 - 0
16977 - 237
16978 - 176
16979 - 235
16980 - 1
16981 - 30
16982 - 0
16983 - 9
16984 - 235
16985 - 193
16986 - 16
16987 - 241
16988 - 201
16989 - 205
16990 - 166
16991 - 64
16992 - 62
16993 - 255
16994 - 237
16995 - 91
16996 - 60
16997 - 64
16998 - 205
16999 - 133
17000 - 66
17001 - 17
17002 - 154
17003 - 64
17004 - 205
17005 - 31
17006 - 66
17007 - 14
17008 - 30
17009 - 62
17010 - 62
17011 - 237
17012 - 71
17013 - 6
17014 - 255
17015 - 16
17016 - 254
```

```
17017 - 61      17038 - 203
17018 - 61      17039 - 37
17019 - 32      17040 - 203
17020 - 246     17041 - 20
17021 - 13      17042 - 203
17022 - 32      17043 - 37
17023 - 241     17044 - 203
17024 - 62      17045 - 20
17025 - 30      17046 - 6
17026 - 237     17047 - 0
17027 - 71      17048 - 74
17028 - 201     17049 - 9
17029 - 38      17050 - 75
17030 - 0       17051 - 9
17031 - 106     17052 - 237
17032 - 203     17053 - 75
17033 - 37      17054 - 12
17034 - 203     17055 - 64
17035 - 37      17056 - 3
17036 - 203     17057 - 9
17037 - 37      17058 - 201
```

Tips + Tricks

Paddle am Commodore 64

Viele Besitzer eines Commodore 64 wissen nicht, daß ihr Computer vier analoge Eingänge besitzt. Diese eignen sich für fast alle handelsüblichen Paddles. Paddles bestehen in der Regel nur aus einem Potentiometer sowie einem Feuerknopf. Je nach Stellung des Reglers wird eine entsprechend große Zahl (0 bis 255) vom Computer ermittelt. Dem Basic-Programmierer ist jedoch leider das Auslesen der Paddles versagt. Das folgende Maschinenprogramm hilft nun, diesen Mangel zu beheben:

```
60000 DATA 120,169,128,32,236,207,142,60,3,140
60001 DATA 61,3,173,0,220,41,12,141,159,2,169
60002 DATA 64,32,236,207,142,62,3,140,63,3,173
60003 DATA 1,220,41,12,141,160,2,169,255,141,2
60004 DATA 220,88,96,141,0,220,9,192,141,2,220
60005 DATA 162,0,220,9,192,141,2,220,162,0,202
60006 DATA 208,253,174,25,212,172,26,212,96
60007 FOR I=53182 TO 53247 : READ D : POKE I,D : NEXT I
60008 PRINT PEEK(830) : REM PADDLE 1 AM PORT 1
60009 PRINT PEEK(831) : REM PADDLE 2 AM PORT 1
60010 PRINT PEEK(672) : REM TASTEN VOM PADDLEPAAR A
60011 PRINT PEEK(828) : REM PADDLE 1 AM PORT 2
60012 PRINT PEEK(829) : REM PADDLE 2 AM PORT 2
60013 PRINT PEEK(671) : REM TASTEN VOM PADDLEPAAR B
```

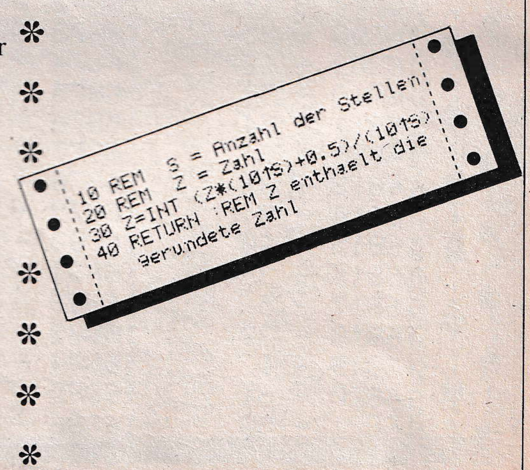
Reset ohne Programmverlust für C 64

Beim Programmieren kommt es häufig vor, daß der Rechner „abstürzt“.
Unter einem Absturz versteht man, daß der Rechner in einer Schleife hängt und auf keine Taste mehr reagiert. Dies kommt meistens dann vor, wenn versehentlich ein falscher POKE-Befehl eingegeben wurde.
Ist der Rechner auch über die Tasten STOP/RESTORE nicht mehr aktivierbar, so hilft nur ein Hardware-Reset.
Um diesen zu erreichen, verbindet man kurzzeitig die Kontakte 1 und 3 des USER-Ports (Computronic Ausgabe 9/84 ausführlich beschrieben).
Leider werden durch einen Reset auch die Basic-Pointer verstellt. Das Programm scheint gelöscht.
Das hier vorgestellte Basic-Programm schreibt nach dem Start ein kleines Maschinenprogramm Namens „HELP“ auf die Diskette.
Sollte Ihr Rechner wieder einmal streiken, so führen Sie einen RESET aus und laden das File HELP mit:
LOAD „HELP“,8,1
Nun starten Sie dieses Programm mit SYS 525 und schon ist Ihr Programm wieder im Speicher. Im übrigen können Sie Help auch nach einem NEW benutzen.

```
1 REM      HELP-FILE ERSTELLER
2 :
10 FORU = 525 TO 577
20 READ DA:POKE U,DA:NEXTU
100 DATA 160,3,200,177,43,208,251
110 DATA 200,200,152,160,0,145,43
120 DATA 165,44,200,145,43,133,60
130 DATA 160,0,132,59,162,0,200,208
140 DATA 2,230,60,177,59,208,245,232
150 DATA 224,3,208,242,200,208,2,230
160 DATA 60,132,45,164,60,132,46,96
200 POKE 43,13:POKE 44,2
210 POKE 45,66:POKE 46,2
220 CLR
230 SAVE "HELP",8
```

Runden von Zahlen

Viele Freaks wissen eigentlich nicht, daß Computer nur sehr ungenau rechnen können. Dies macht sich insbesondere bei Zahlen mit Nachkommastellen bemerkbar. Mit dem nachfolgenden Programm kann man Zahlen auf beliebige Stellen runden und somit die Ungenauigkeit etwas ausgleichen.
Das Programm kann als Unterprogramm für fast alle Basic-Computer verwendet werden.



```
10 REM  S = Anzahl der Stellen
20 REM  Z = Zahl
30 Z=INT (Z*(10↑S)+0.5)/(10↑S)
40 RETURN :REM Z enthaelt die
   gerundete Zahl
```

MANIAC!

für Atari

Ein Spiel mit verschiedenen Bildern, geschrieben auf dem Atari 800XL

Stellen Sie sich vor, Sie besitzen ein Haus mit zehn Räumen, und eines Morgens finden Sie in jedem der zehn Räume sechs Marsmännchen auf. Zum Glück stellen Sie fest, daß diese Männchen nicht bösartig sind, aber Sie haben Ihre Aufzüge, die Sie normalerweise von Etage zu Etage bringen, so präpariert, daß sie sich nach einmaligem Gebrauch selbst zerstören. Dies erschwert Ihnen das Auffinden dieser fremden Wesen natürlich enorm. Hinzu kommt, daß Sie einen Raum erst wieder verlassen können, wenn Sie alle diese Wesen eingesammelt haben. Haben Sie dies geschafft, so müssen Sie die Tür rechts unten in der Ecke berühren, um in den nächsten Raum zu kommen. Ist Ihre Zeit abgelaufen und steht auf Null, so verlieren Sie eines Ihrer drei Leben. Gespielt wird mit Joystick in Port 1.

```
10 REM ***********************
20 REM ** M  A  N  I  A  C **
30 REM **   COPYRIGHT BY    **
40 REM ** BERND "BERNHARD"  **
50 REM **      THOMAS       **
60 REM **    5800 HAGEN 1   **
70 REM **  HUMPERTSTR.17    **
80 REM ** TELE.02331/82857 **
90 REM ***********************
91 GRAPHICS 18:POSITION 4,5:? #6;"PLEASE WAIT"
99 REM ZEICHEN DEFIENIEREN
100 PAGE=PEEK(106)-8:CHS=PAGE*256
110 FOR I=128 TO 471:POKE CHS+I,PEEK(57344+I):NEXT I:RESTORE 150
120 READ C:IF C<>-1 THEN FOR I=0 TO 7:READ A:POKE CHS+C*8+I,A:SOUND 0,A,10,15:NE
XT I:GOTO 120
150 DATA 1,255,255,153,102,102,153,255,255
155 DATA 2,247,247,63,63,252,252,239,239
160 DATA 3,255,36,255,60,231,231,165,255
165 DATA 4,195,255,195,255,195,255,195,255
170 DATA 5,0,195,195,60,90,126,66,60
175 DATA 6,255,211,171,213,211,173,211,255
180 DATA 26,8,16,60,248,56,40,68,130
185 DATA 27,16,8,60,31,28,20,34,65
190 DATA 7,54,255,54,255,54,255,54,255
195 DATA 28,68,18,66,16,156,126,255,255,-1
196 SOUND 0,0,0,0
200 POKE 559,34:POKE 77,0:GRAPHICS 17:POKE 756,PAGE
210 COLOR 160+1:FOR I=0 TO 19:PLOT I,0:PLOT I,23:NEXT I
220 COLOR 160+2:FOR I=1 TO 22:PLOT 0,I:PLOT 19,I:NEXT I
230 POSITION 4,2:? #6;"M A N I A C"
240 POSITION 8,4:? #6;"BY"
250 POSITION 3,6:? #6;"BERND  THOMAS"
260 X1=1:X2=17:XR1=0:XR2=0
265 COLOR 26:PLOT 9,12
270 COLOR 37:PLOT X1,12:COLOR 37:PLOT X2,12
275 XR1=1:XR2=-1
276 FOR K=0 TO 100:NEXT K
280 COLOR 32:PLOT X1,12:PLOT X2,12
285 SOUND 0,X1+50,10,15:SOUND 1,X2+80,10,15:SOUND 0,0,0,0:SOUND 1,0,0,0
290 X1=X1+XR1:X2=X2+XR2
291 IF X1=9 OR X2=9 THEN COLOR 28:PLOT 9,12
295 IF X1=9 OR X2=9 THEN SOUND 0,150,8,15:FOR I=0 TO 150:NEXT I:SOUND 0,0,0,0
300 IF X1=9 OR X2=9 THEN GOTO 340
310 GOTO 270
340 COLOR 32:PLOT 9,12
341 POSITION 4,12:? #6;"M A N I A C"
345 POSITION 4,16:? #6;"PRESS START"
346 FOR K=0 TO 50:NEXT K
350 SOUND 0,60,10,15
355 FOR I=0 TO 100:NEXT I
360 POSITION 4,16:? #6;"PRESS START"
370 SOUND 0,0,0,0
371 FOR K=0 TO 50:NEXT K
375 POSITION 4,16:? #6;"PRESS START"
376 FOR K=0 TO 50:NEXT K
380 IF PEEK(53279)<>6 THEN 345
381 GRAPHICS 17:POSITION 0,10:? #6;"SPIELANLEITUNG (J/N)"
382 POSITION 5,12:? #6;"SELECT=JA":POSITION 5,14:? #6;"OPTION=NEIN":FOR U=0 TO 2
00:NEXT U
```

```
383 IF PEEK(53279)=3 THEN GOTO 386
384 IF PEEK(53279)=5 THEN GOTO 9000
385 GOTO 383
386 GRAPHICS 17:POKE 756,PAGE:POKE 708,15:POKE 709,24
399 REM VARIABELN
400 LIV=3:POI=0:TI=300:IT=0:RO=0:X=1:Y=22:F=0
405 GOSUB 410
407 GOTO 500
410 POSITION 1,0:? #6;"TIME ":POSITION 11,0:? #6;"LIVES":POSITION 1,1:? #6;"POIN
TS":POSITION 13,1:? #6;"ROOM"
411 RETURN
499 REM SPRUNG BILDERAUFBAU
500 GOTO 20000
501 COLOR 27:PLOT 1,22
599 REM HAUPTSCHLEIFE
600 TI=TI-1
601 IF TI=-1 THEN 2000
605 ST=STICK(0)
606 XR=0:YR=0
607 F=0
610 POSITION 6,0:? #6;TI;"  "
630 IF ST=7 THEN XR=1:CHR=27
640 IF ST=11 THEN XR=-1:CHR=26
645 IF XR=0 AND YR=0 THEN 600
650 LOCATE X+XR,Y+YR,L
660 IF L=4 THEN GOTO 3000
665 IF L=166 OR L=161 OR L=162 OR L=163 THEN GOTO 600
670 IF L=7 AND IT=6 THEN GOTO 5000
675 IF L=39 AND IT<6 THEN GOTO 600
690 IF L=37 THEN 6000
700 LOCATE X+XR,Y+1,B
710 IF B=32 THEN GOTO 4000
720 IF B<>32 THEN SOUND 0,30,10,15:SOUND 0,0,0,0
723 COLOR 32:PLOT X,Y
725 X=X+XR:Y=Y+YR
730 COLOR CHR:PLOT X,Y
740 GOTO 600
2000 REM
2010 COLOR 32:PLOT X,Y
2011 COLOR 28:PLOT X+XR,Y+YR
2020 LIV=LIV-1
2030 FOR K=0 TO 255 STEP 5.5
2035 SOUND 0,K,10,10:SOUND 1,K,8,4:SOUND 2,K,10,15
2040 NEXT K
2050 SOUND 0,0,0,0:SOUND 1,0,0,0:SOUND 2,0,0,0
2055 IF LIV=0 THEN GOTO 8000
2060 TI=300
2065 COLOR 32:PLOT X+XR,Y+YR
2066 PLOT X,Y
2070 GOTO 20005
3000 COLOR 32:PLOT X+XR,Y+YR:COLOR 32:PLOT X,Y
3010 FOR K=120 TO 255 STEP 5
3020 SOUND 0,K,10,15:SOUND 1,K-30,10,15:SOUND 2,K-70,10,15:NEXT K
3030 SOUND 0,0,0,0:SOUND 1,0,0,0:SOUND 2,0,0,0
3040 Y=Y-2
3050 COLOR CHR+32:PLOT X+XR,Y:FOR K=0 TO 10:NEXT K:COLOR CHR+160:PLOT X+XR,Y:FOR
 K=0 TO 10:NEXT K
3061 COLOR CHR:PLOT X+XR,Y
3062 COLOR 32:PLOT X+XR,Y
3070 GOTO 725
4000 COLOR 32:PLOT X,Y
4001 LOCATE X+XR,Y+1,M
4002 IF M=7 THEN GOTO 5000
4003 IF M<>32 THEN SOUND 0,0,0,0:GOTO 4100
4004 F=F+1
4030 Y=Y+1
4040 COLOR 32:PLOT X+XR,Y-1
4050 COLOR CHR:PLOT X+XR,Y
4060 FOR T=0 TO 1:NEXT T
4065 SOUND 0,Y*5+10,14,8:SOUND 0,0,0,0
4070 GOTO 4001
4100 IF F<6 THEN GOTO 725
4106 IF M=37 THEN GOTO 6000
4110 IF F>6 THEN GOTO 2000
4120 GOTO 725
5000 IF IT<>6 THEN GOTO 600
5010 COLOR 32:PLOT X,Y:PLOT X+1,Y
5020 COLOR 39:PLOT 18,22
```

```
5030 RESTORE 5200
5040 READ A:IF A=-1 THEN GOTO 5300
5050 FOR T=0 TO 10:NEXT T
5060 SOUND 0,A,10,13
5070 GOTO 5040
5200 DATA 61,61,48,48,43,43,32,38,48,61,61,61,46,46,41,41,31,36,46,46,48,54,34,4
1,48,46
5300 IT=0
5305 SOUND 0,0,0,0
5400 GOTO 20000
6000 POI=POI+5
6001 POSITION 8,1:? #6;POI
6002 IT=IT+1
6003 COLOR 32:PLOT X+XR,Y+YR
6010 IF IT=6 THEN COLOR 7:PLOT 18,22
6020 RESTORE 6100
6030 READ A:IF A=-1 THEN GOTO 6200
6040 SOUND 0,A,10,14:SOUND 0,A-50,10,13
6050 SOUND 0,0,0,0:SOUND 1,0,0,0
6060 GOTO 6030
6100 DATA 96,91,85,81,76,72,68,64,96,91,85,81,76,72,68,64,96,85,76,68,96,81,68,9
6,76,64,-1
6200 SOUND 0,0,0,0
6230 GOTO 600
7000 GRAPHICS 17:POKE 756,PAGE
7010 COLOR 162:FOR I=0 TO 19:PLOT I,0:PLOT I,23:NEXT I
7020 COLOR 163:FOR I=1 TO 22:PLOT 0,I:PLOT 19,I:NEXT I
7025 POSITION 4,2:? #6;"M A N I A C"
7030 POSITION 5,5:? #6;"HERZLICHEN"
7040 POSITION 4,7:? #6;"GLUECKWUNSCH "
7041 POSITION 4,9:? #6;"COPYRIGHT BY"
7045 POSITION 2,12:? #6;"******  *******":POSITION 2,13:? #6;"*     *    *"
7046 POSITION 2,14:? #6;"*   *   *"
7047 POSITION 2,15:? #6;"*******   *"
7048 POSITION 2,16:? #6;"*    *   *"
7049 POSITION 2,17:? #6;"*    *   *"
7050 POSITION 2,18:? #6;"******* *  * *"
7498 RESTORE 7504
7499 READ A,B,C,D:IF A=-1 THEN RESTORE 7504
7500 IF A=-1 THEN GOTO 7600
7501 SOUND 1,B,10,6:SOUND 0,A,10,15:SOUND 2,C,10,6:SOUND 3,D,10,6
7502 FOR P=1 TO 25:NEXT P
7503 GOTO 7499
7504 REM T A K T  1
7505 DATA 53,108,128,162,53,108,128,162
7506 DATA 81,108,128,162,72,108,128,162
7507 DATA 64,144,144,144,60,144,144,144
7508 REM TAKT 2
7509 DATA 53,128,128,128,53,128,128,128
7510 DATA 81,128,128,128,81,128,128,128
7511 DATA 81,128,128,128
7512 DATA 81,128,128,128
7513 REM TAKT 3
7514 DATA 47,121,121,121,47,121,121,121
7515 DATA 60,121,121,121,53,121,121,121
7516 DATA 47,121,121,121,42,121,121,121
7517 REM TAKT 4
7518 DATA 40,128,128,128,40,128,128,128
7519 DATA 81,128,128,128,81,128,128,128
7520 DATA 81,128,128,128,81,128,128,128
7521 REM TAKT 5
7522 DATA 60,144,144,144,60,144,144,144
7523 DATA 53,144,144,144,60,144,144,144
7524 DATA 64,144,144,144,72,144,144,144
7525 REM TAKT 6
7526 DATA 64,162,162,162,64,162,162,162
7527 DATA 60,162,162,162,64,162,162,162
7528 DATA 72,162,162,162,81,162,162,162
7529 REM TAKT 7
7530 DATA 85,108,108,108,85,108,108,108
7531 DATA 81,128,128,128,72,128,128,128
7532 DATA 64,162,162,162,81,162,162,162
7533 REM TAKT 8
7534 DATA 64,108,108,108,72,108,108,108
7535 DATA 72,217,217,217,72,121,121,121
7536 DATA 72,128,128,128,72,144,144,144
7537 REM TAKT 9
7538 DATA 53,128,128,128,53,128,128,128
```

```
7539 DATA 81,128,128,128,72,128,128,128
7540 DATA 64,144,144,144,60,144,144,144
7541 REM TAKT 10
7542 DATA 53,162,162,162,53,162,162,162
7543 DATA 81,128,128,128,81,128,128,128
7544 DATA 81,162,162,162
7545 DATA 81,162,162,162
7546 REM TAKT 11
7547 DATA 47,121,121,121,47,121,121,121
7548 DATA 60,121,121,121,53,121,121,121
7549 DATA 47,121,121,121,42,121,121,121
7550 REM TAKT 12
7551 DATA 40,128,128,128,40,128,128,128
7552 DATA 81,121,121,121,81,128,128,128
7553 DATA 81,144,144,144,81,162,162,162
7554 REM TAKT 13
7555 DATA 60,144,144,144,60,144,144,144
7556 DATA 53,144,144,144,60,144,144,144
7557 DATA 64,173,173,173,72,173,173,173
7558 REM TAKT 14
7559 DATA 64,162,162,162,64,162,162,162
7560 DATA 60,162,162,162,64,162,162,162
7561 DATA 72,128,128,128,81,128,128,128
7561 DATA 72,128,128,128,81,128,128,128
7562 REM TAKT 15
7563 DATA 72,121,121,121,72,121,121,121
7564 DATA 64,108,108,108,72,108,108,108
7565 DATA 81,217,217,217,85,217,217,217
7566 REM TAKT 16
```

```
7567 DATA 81,162,162,162,81,162,162,162,81,162,162,162,81,162,162,162
7568 DATA 81,162,162,162,81,162,162,162
7569 DATA -1,0,0,0
7600 POSITION 2,20:? #6;"% PRESS START %"
7601 SOUND 0,0,0,0:SOUND 1,0,0,0:SOUND 2,0,0,0:SOUND 3,0,0,0
7604 IF PEEK(53279)<>6 THEN GOTO 7600
7605 GOTO 8000
8000 GRAPHICS 17:POKE 756,PAGE
8010 COLOR 37:FOR I=0 TO 19:PLOT I,0:PLOT I,23:NEXT I:FOR I=1 TO 22:PLOT 0,I:PLO
T 19,I:NEXT I
8020 POSITION 4,2:? #6;"M A N I A C"
8030 POSITION 4,4:? #6;"COPYRIGHT by"
8035 POSITION 4,6:? #6;"BERND THOMAS"
8040 POSITION 2,10:? #6;"PUNKTE ";POI
8045 IF POI>=HS THEN HS=POI
8046 IF RO>RH THEN RH=RO
8050 POSITION 2,14:? #6;"HIGH SCORE ";HS
8060 POSITION 2,12:? #6;"RAUM  ";RO
8065 POSITION 2,16:? #6;"HIGH ROOM ";RH
8070 POSITION 6,18:? #6;"NOCHMAL ? "
8075 POSITION 4,20:? #6;" PRESS START"
8080 IF PEEK(53279)<>6 THEN GOTO 8080
8090 GOTO 200
9000 GRAPHICS 0:SETCOLOR 2,0,0
9010 ? "          M  A  N  I  A  C"
9020 ? :? :? "SIE SIND EIN KLEINES MAENNCHEN,DAS EINHAUS MIT ZEHN RAEUMEN BESITZ
T. "
9030 ? "IN JEDEM DIESER ZEHN RAEUME VERSTECKENSICH 6 WAHNSINNIGE,DIE SICH ALS MA
RS- MAENNCHEN VERKLEIDET ";
9040 ? "HABEN."
9045 ? "SIE MUESSEN DIESE EINSAMMELN.          UM VON EINER ETAGE ZUR ANDEREN ZUK
OMM-EN STEHEN IHNEN AUFZUEGE"
9050 ? "ZUR VERFUEGUNG."
9055 ? "DIE SIE ABER NUR EINMAL GEBRAUCHEN    KOENNEN,WEIL SIE SICH DANACH AUFLO
ESEN"
9060 ? "HABEN SIE ALLE WAHNSINNIGEN EINGESAMM-ELT,SO MUESSEN SIE DIE GITTERTUER
UNT-EN RECHTS IN DER ECKE ";
9065 ? "BERUEHREN."
9070 ? "DIES ALLES MUSS IN EINER BESTIMMTEN   ZEIT PASSIEREN,SONST VERLIEREN SIE
 EINLEBEN."
9080 ? "PRESS START"
9090 IF PEEK(53279)<>6 THEN GOTO 9090
9100 GRAPHICS 17:POKE 756,PAGE
9110 POSITION 0,0:? #6;"ZEICHENERKLAERUNG"
9115 POSITION 0,1:? #6;"-----------------"
9120 POSITION 0,5:? #6;CHR$(161);" BADEN"
9121 POSITION 0,7:? #6;CHR$(162);" BADEN"
9122 POSITION 0,9:? #6;CHR$(163);" BADEN"
9123 POSITION 0,11:? #6;CHR$(166);" WAND"
9124 POSITION 0,13:? #6;CHR$(26);" ";CHR$(27);" MAENNCHEN"
9125 POSITION 0,15:? #6;CHR$(28);" TOTES MAENNCHEN"
9126 POSITION 0,17:? #6;CHR$(4);" AUFZUG"
9127 POSITION 0,19:? #6;CHR$(39);" gittertuer"
9128 POSITION 0,21:? #6;CHR$(37);" wahnsinniger"
9129 POSITION 0,23:? #6;"PRESS OPTION"
9130 IF PEEK(53279)<>3 THEN GOTO 9130
9135 GOTO 386
19999 REM RAUEME BAUEN
20000 IF RO=10 THEN GOTO 7000
20001 GRAPHICS 17:POKE 756,PAGE
20003 GOSUB 410:RO=RO+1:POSITION 18,1:? #6;RO
20004 IF RO<>1 THEN ZU=TI:IF RO<>1 THEN POI=POI+ZU
20005 TI=300:POSITION 6,0:? #6;TI
20006 POSITION 8,1:? #6;POI:POSITION 17,0:? #6;LIV
20007 COLOR 160+6:FOR I=0 TO 19:PLOT I,2:PLOT I,23:NEXT I:FOR H=2 TO 23:PLOT 0,H
:PLOT 19,H:NEXT H
20008 COLOR 39:PLOT 18,22
20009 X=1:Y=22:IT=0
20010 GOSUB 20000+RO*100
20020 GOTO 501
20100 COLOR 161:PLOT 1,21:DRAWTO 14,21:PLOT 3,19:DRAWTO 17,19:PLOT 6,17:DRAWTO 1
7,17:PLOT 1,15:DRAWTO 8,15
20110 PLOT 2,13:DRAWTO 18,13:PLOT 8,11:DRAWTO 16,11:PLOT 5,9:DRAWTO 18,9:PLOT 7,
7:DRAWTO 18,7
20115 COLOR 4:PLOT 14,22:PLOT 3,20:PLOT 13,18:PLOT 8,16:PLOT 3,14:PLOT 8,12:PLOT
 12,10:PLOT 9,8
20120 COLOR 37:PLOT 1,20:PLOT 9,18:PLOT 2,14:PLOT 12,12:PLOT 11,10:PLOT 7,6
20123 POKE 710,4:POKE 711,36
```

```
20130 RETURN
20200 COLOR 162:PLOT 1,21:DRAWTO 12,21:PLOT 1,19:DRAWTO 6,19:PLOT 9,19:DRAWTO 14
,19:PLOT 5,17:DRAWTO 15,17
20210 PLOT 2,15:DRAWTO 14,15:PLOT 3,13:DRAWTO 5,13:PLOT 8,13:DRAWTO 13,13:PLOT 5
,11:DRAWTO 8,11
20215 PLOT 8,9:DRAWTO 12,9:PLOT 14,9:DRAWTO 18,9:PLOT 10,7:DRAWTO 16,7
20220 COLOR 4:PLOT 6,22:PLOT 3,20:PLOT 10,20:PLOT 9,18:PLOT 12,18:PLOT 5,16:PLOT
 11,16:PLOT 3,14:PLOT 10,14
20225 PLOT 5,12:PLOT 8,10:PLOT 11,8
20230 COLOR 166:PLOT 8,14:PLOT 10,16:PLOT 8,12
20235 COLOR 37:PLOT 2,18:PLOT 11,18:PLOT 13,14:PLOT 6,10:PLOT 16,8:PLOT 10,12
20240 COLOR 162:PLOT 16,21
20243 POKE 710,166:POKE 711,196
20245 RETURN
20300 COLOR 161:PLOT 1,21:DRAWTO 10,21:PLOT 12,20:DRAWTO 17,20:PLOT 2,19:PLOT 3,
19:PLOT 5,19:DRAWTO 10,19
20305 PLOT 2,17:DRAWTO 11,17
20310 COLOR 162:PLOT 2,15:PLOT 3,15:PLOT 6,15:DRAWTO 9,15:PLOT 4,13:DRAWTO 12,13
:PLOT 6,11:DRAWTO 16,11
20315 COLOR 163:PLOT 7,9:DRAWTO 16,9:PLOT 9,7:DRAWTO 17,7:PLOT 8,5:DRAWTO 18,5
20320 COLOR 166:PLOT 6,22:PLOT 10,20:PLOT 8,18:PLOT 5,16
20325 COLOR 4:PLOT 2,18:PLOT 3,20:PLOT 4,22:PLOT 5,20:PLOT 6,18:PLOT 6,14:PLOT 8
,16:PLOT 9,20:PLOT 9,12
20330 PLOT 10,18:PLOT 11,8:PLOT 14,10:PLOT 14,6
20335 COLOR 37:PLOT 2,20:PLOT 15,19:PLOT 7,16:PLOT 2,14:PLOT 8,8:PLOT 12,4
20340 RETURN
20400 COLOR 163:PLOT 7,21:DRAWTO 9,21:PLOT 3,19:DRAWTO 14,19:PLOT 6,17:DRAWTO 16
,17:PLOT 7,15:DRAWTO 17,15
20405 COLOR 162:PLOT 8,13:DRAWTO 17,13:PLOT 9,11:DRAWTO 16,11
20410 COLOR 161:PLOT 12,9:DRAWTO 14,9:PLOT 12,7:DRAWTO 14,7:PLOT 12,5:DRAWTO 14,
5
20415 COLOR 166:PLOT 8,18:PLOT 13,14:PLOT 12,12
20420 COLOR 4:PLOT 8,22:PLOT 7,20:PLOT 9,20:PLOT 13,18:PLOT 15,16:PLOT 9,14:PLOT
 16,14:PLOT 14,12:PLOT 13,10
20425 PLOT 14,8:PLOT 11,16:PLOT 13,6
20430 COLOR 37:PLOT 5,18:PLOT 11,18:PLOT 12,14:PLOT 14,14:PLOT 10,12:PLOT 12,4
20435 POKE 710,4:POKE 711,36
20440 RETURN
20500 COLOR 162:PLOT 1,21:DRAWTO 5,21:PLOT 9,21:DRAWTO 13,21:PLOT 1,19:DRAWTO 5
19:PLOT 9,19:DRAWTO 12,19
20505 PLOT 1,17:DRAWTO 7,17:PLOT 11,17:DRAWTO 15,17:PLOT 2,15:DRAWTO 7,15:PLOT
2,15:DRAWTO 15,15
20510 COLOR 163:PLOT 2,13:DRAWTO 8,13:PLOT 10,13:PLOT 11,13:PLOT 13,13:DRAWTO 1
,13:PLOT 1,11:DRAWTO 8,11
20515 PLOT 13,11:DRAWTO 16,11:PLOT 3,9:DRAWTO 9,9:PLOT 13,9:DRAWTO 17,9:PLOT 7,
:DRAWTO 17,7
20520 PLOT 8,5:DRAWTO 14,5
20525 COLOR 166:PLOT 6,22:PLOT 4,16:PLOT 2,14:PLOT 7,14:PLOT 2,12:PLOT 8,12:PLO
 12,22
20530 COLOR 4:PLOT 2,22:PLOT 2,18:PLOT 3,20:PLOT 3,16:PLOT 3,12:PLOT 3,10:PLOT
,14:PLOT 5,18:PLOT 9,20
20535 PLOT 11,22:PLOT 11,18:PLOT 11,6:PLOT 13,16:PLOT 13,12:PLOT 13,8:PLOT 14,1
:PLOT 15,10
20540 COLOR 37:PLOT 1,10:PLOT 4,18:PLOT 8,8:PLOT 9,4:PLOT 14,10:PLOT 11,12
20545 POKE 710,150:POKE 711,164
20550 RETURN
20600 COLOR 163:PLOT 5,21:DRAWTO 13,21:PLOT 1,19:DRAWTO 6,19:PLOT 11,19:DRAWTO 1
4,19:PLOT 1,17:DRAWTO 4,17
20605 PLOT 13,17:PLOT 14,17:PLOT 1,15:DRAWTO 4,15:PLOT 12,15:DRAWTO 17,15:PLOT 1
,13:PLOT 2,13:PLOT 11,13
20606 DRAWTO 14,13:COLOR 162:PLOT 1,11:DRAWTO 15,11:PLOT 3,9:DRAWTO 16,9:PLOT 3,
7:DRAWTO 6,7:PLOT 14,7
20607 DRAWTO 16,7:PLOT 1,5:DRAWTO 4,5:PLOT 6,5:DRAWTO 17,5
20610 COLOR 4:PLOT 8,22:PLOT 5,20:PLOT 13,20:PLOT 2,18:PLOT 14,18:PLOT 4,16:PLOT
 13,16:PLOT 1,14:PLOT 12,14
20615 PLOT 11,12:PLOT 5,10:PLOT 14,10:PLOT 3,8:PLOT 15,8:PLOT 4,6:PLOT 14,6:PLOT
 16,6
20620 COLOR 37:PLOT 1,18:PLOT 2,12:PLOT 13,12:PLOT 1,4:PLOT 12,4:PLOT 6,6
20625 COLOR 166:PLOT 6,8:PLOT 16,8:PLOT 15,4
20627 POKE 710,4:POKE 711,36
20630 RETURN
20700 COLOR 161:PLOT 1,21:DRAWTO 9,21:PLOT 2,19:DRAWTO 9,19:PLOT 2,17:DRAWTO 9,1
7:PLOT 2,15:DRAWTO 7,15
20705 PLOT 1,13:DRAWTO 11,13:PLOT 3,11:DRAWTO 6,11:PLOT 2,9:DRAWTO 6,9:PLOT 1,7:
DRAWTO 7,7:PLOT 1,5:DRAWTO 3,5
20710 PLOT 7,5:DRAWTO 9,5
20715 COLOR 162:PLOT 11,21:DRAWTO 18,21:PLOT 12,19:DRAWTO 17,19:PLOT 12,17:DRAWT
O 18,17:PLOT 13,15:DRAWTO 17,15
```

```
20720 PLOT 13,13:DRAWTO 17,13:PLOT 10,11:DRAWTO 18,11:PLOT 11,9:DRAWTO 15,9:PLOT
 9,7:DRAWTO 18,7
20725 PLOT 11,5:DRAWTO 17,5
20730 COLOR 166:PLOT 9,20:PLOT 6,18:PLOT 15,18:PLOT 2,16:PLOT 6,16:PLOT 15,14:PL
OT 6,12:PLOT 15,12:PLOT 3,10
20735 PLOT 6,10:PLOT 11,10:PLOT 6,8:PLOT 3,6:PLOT 13,6:PLOT 15,6
20740 COLOR 4:PLOT 3,22:PLOT 12,22:PLOT 4,20:PLOT 8,20:PLOT 13,20:PLOT 16,20:PLO
T 5,18:PLOT 12,18:PLOT 4,16
20745 PLOT 14,16:PLOT 17,16:PLOT 3,14:PLOT 13,14:PLOT 16,14:PLOT 5,12:PLOT 14,12
:PLOT 4,10:PLOT 12,10
20750 PLOT 2,8:PLOT 11,8:PLOT 14,8:PLOT 1,6:PLOT 9,6:PLOT 12,6:PLOT 16,6
20755 COLOR 37:PLOT 3,18:PLOT 7,18:PLOT 17,18:PLOT 17,12:PLOT 17,6:PLOT 7,6
20780 POKE 710,140:POKE 711,20
20790 RETURN
20800 COLOR 161:PLOT 1,21:DRAWTO 17,21:PLOT 1,19:DRAWTO 4,19:PLOT 6,19:DRAWTO 10
,19:PLOT 13,19:DRAWTO 17,19
20805 PLOT 1,17:PLOT 2,17:PLOT 4,17:DRAWTO 6,17:PLOT 9,17:DRAWTO 13,17:PLOT 15,1
7:DRAWTO 17,17
20806 PLOT 1,15:PLOT 2,15:PLOT 5,15:DRAWTO 9,15:PLOT 12,15:DRAWTO 16,15:PLOT 1,1
3:PLOT 2,13
20807 PLOT 5,13:DRAWTO 9,13:PLOT 12,13:DRAWTO 16,13:PLOT 1,11:DRAWTO 5,11:PLOT 8
,11:DRAWTO 13,11
20808 PLOT 1,9:DRAWTO 11,9:PLOT 15,9:DRAWTO 17,9:PLOT 2,7:DRAWTO 7,7:PLOT 10,7:D
RAWTO 15,7
20809 PLOT 2,5:DRAWTO 5,5:PLOT 7,5:DRAWTO 11,5:PLOT 13,5:DRAWTO 16,5
20810 COLOR 166:PLOT 10,10:PLOT 5,4
20815 COLOR 4:PLOT 17,22:PLOT 3,20:PLOT 8,20:PLOT 14,20:PLOT 2,18:PLOT 4,18:PLOT
 6,18:PLOT 10,18:PLOT 13,18
20816 PLOT 16,18:PLOT 1,16:PLOT 5,16:PLOT 9,16:PLOT 12,16:PLOT 15,16:PLOT 2,14:P
LOT 7,14:PLOT 14,14
20817 PLOT 5,12:PLOT 9,12:PLOT 12,12:PLOT 4,8:PLOT 6,8:PLOT 10,8:PLOT 2,6:PLOT 7
,6:PLOT 13,6:PLOT 15,6
20818 PLOT 1,10
20820 COLOR 37:PLOT 1,12:PLOT 4,4:PLOT 8,10:PLOT 11,10:PLOT 13,14:PLOT 14,6
20830 RETURN
20900 COLOR 161:PLOT 1,21:DRAWTO 12,21:PLOT 14,21:DRAWTO 18,21:PLOT 1,19:DRAWTO
10,19:PLOT 13,19:DRAWTO 18,19
20905 COLOR 162:PLOT 2,17:DRAWTO 9,17:PLOT 12,17:PLOT 15,17:DRAWTO 18,17:PLOT 3,
15:DRAWTO 9,15:PLOT 11,15
20910 PLOT 16,15:DRAWTO 18,15:PLOT 1,13:DRAWTO 8,13:PLOT 12,13:DRAWTO 17,13:PLOT
 2,11:DRAWTO 7,11
20915 PLOT 14,11:DRAWTO 18,11:PLOT 2,9:DRAWTO 8,9:PLOT 10,9:DRAWTO 17,9:PLOT 1,7
:DRAWTO 9,7:PLOT 11,7
20920 DRAWTO 18,7:COLOR 163:PLOT 3,5:DRAWTO 9,5:PLOT 12,5:DRAWTO 14,5:PLOT 16,5:
PLOT 17,5
20925 COLOR 166:PLOT 12,22:PLOT 15,18:PLOT 9,16:PLOT 8,14:PLOT 14,12:PLOT 7,12:P
LOT 7,10
20930 COLOR 4:PLOT 4,22:PLOT 15,22:PLOT 2,20:PLOT 17,20:PLOT 4,18:PLOT 16,18:PLO
T 6,16:PLOT 18,16:PLOT 4,14
20935 PLOT 16,14:PLOT 2,12:PLOT 5,10:PLOT 9,8:PLOT 17,8:PLOT 5,6:PLOT 8,6:PLOT 1
4,6:PLOT 16,6:PLOT 17,12
20940 COLOR 37:PLOT 7,18:PLOT 3,16:PLOT 14,18:PLOT 15,12:PLOT 1,6:PLOT 17,4
20945 POKE 710,34:POKE 711,46
20950 RETURN
21000 COLOR 163:PLOT 1,21:DRAWTO 8,21:PLOT 1,19:DRAWTO 4,19:PLOT 6,19:DRAWTO 8,1
9:PLOT 15,19:DRAWTO 17,19
21005 PLOT 1,17:PLOT 2,17:PLOT 7,17:DRAWTO 9,17:PLOT 18,17:PLOT 1,15:PLOT 2,15:P
LOT 4,15:DRAWTO 6,15
21010 PLOT 8,15:DRAWTO 16,15:COLOR 162:PLOT 1,13:DRAWTO 4,13:PLOT 7,13:DRAWTO 12
,13:PLOT 14,13:PLOT 15,13
21015 PLOT 17,13:PLOT 1,11:DRAWTO 7,11:PLOT 10,11:DRAWTO 12,11:PLOT 14,11:DRAWTO
 16,11
21020 COLOR 161:PLOT 3,9:DRAWTO 6,9:PLOT 10,9:DRAWTO 12,9:PLOT 14,9:PLOT 15,9:PL
OT 18,9:PLOT 2,7:PLOT 3,7
21025 PLOT 5,7:DRAWTO 7,7:PLOT 9,7:PLOT 11,7:DRAWTO 14,7:PLOT 16,6:PLOT 17,6:PLO
T 2,5:DRAWTO 8,5:PLOT 11,5
21030 DRAWTO 14,5:PLOT 16,4:DRAWTO 18,4
21035 COLOR 166:PLOT 6,18:PLOT 15,18:PLOT 7,16:PLOT 10,14:PLOT 9,12:PLOT 7,6:PLO
T 15,6:PLOT 3,18
21040 COLOR 4:PLOT 3,22:PLOT 2,20:PLOT 7,20:PLOT 1,18:PLOT 5,18:PLOT 8,18:PLOT 2
,16:PLOT 9,16:PLOT 1,14
21045 PLOT 8,14:PLOT 11,14:PLOT 15,14:PLOT 7,12:PLOT 10,12:PLOT 14,12:PLOT 4,10:
PLOT 5,10:PLOT 15,10
21050 PLOT 3,8:PLOT 6,8:PLOT 14,8:PLOT 2,6:PLOT 5,6:PLOT 12,6:PLOT 17,5
21055 COLOR 37:PLOT 5,18:PLOT 16,18:PLOT 3,10:PLOT 11,10:PLOT 18,3:PLOT 7,4
21056 COLOR 32:PLOT 5,19
21060 RETURN
```

Fortsetzung von Seite 62

```
3220 LET y=0+15*(y1=20)+9*(y1=15
)+3*(y1=9)
3225 IF y=3 THEN GO SUB 6020
3230 RETURN
4000 RESTORE 4500: FOR n=1 TO 7
4010 READ a,b: BEEP a,b
4020 NEXT n
4030 LET ll=ll+1
4040 FOR n=1 TO 30: PRINT #0;AT
1,12; INK 2;"LEVEL ";ll
4050 PRINT #0;AT 1,12;"
"
4055 BEEP .002,30
4060 NEXT n
4070 IF q1>=8 THEN LET q1=q1-1:
LET q2=q2-1
4080 IF q1<8 AND q1>=5 THEN LET
q1=q1-1
4090 IF q1<5 AND q1>=3 THEN LET
q1=q1-1: LET q2=q2-1: LET q3=q3+
.25
4100 IF q1<3 THEN LET q3=q3+.25:
 LET q1=q1-1
4110 FOR n=1 TO 3
4120 IF c(n)=1 THEN LET p(1)=p(1
)+20
4130 IF c(n)=2 THEN LET p(1)=p(1
)+130
4140 IF c(n)=3 THEN LET p(1)=p(1
)+80
4150 NEXT n
4160 IF c(1)=1 AND c(2)=1 AND c(
3)=1 THEN LET p(1)=p(1)-260
4200 GO SUB 100: GO TO 201
4500 DATA .8,-4,.8,-3,.8,-1,.4,6
,.8,-1,.8,-3,.8,-6
5000 RESTORE 5000: FOR n=USR "a"
 TO USR "o"+7: READ a: POKE n,a:
 NEXT n
5100 DATA 255,0,126,66,126,0,255
,0,169,85,42,170,72,126,255,0
5105 DATA 0,0,0,24,60,126,255,0,
189,60,0,126,126,0,255,0
5110 DATA 24,24,28,16,56,16,40,6
8,24,24,56,8,28,8,20,34
5115 DATA 126,60,24,60,126,255,2
55,126,0,60,126,126,126,126,60,0
5120 DATA 37,74,36,126,219,255,2
31,126,126,255,219,255,102,60,44
,24
5125 DATA 129,195,165,153,153,16
5,195,129,0,102,255,255,126,60,2
4,0
5126 DATA 24,60,126,255,255,126,
24,60,24,126,24,24,24,126,24,24
5127 DATA 0,8,12,254,255,254,12,
8
5130 RETURN
6000 IF ATTR (y,x)=70 THEN FOR n
=5 TO 9: PRINT AT n,x4;" ": NEXT
 n: LET x4=27: GO SUB 650
6005 IF ATTR (4,x)=85 OR ATTR (4
,x)=71 THEN PRINT AT 4,x;" ": GO
 SUB 641
6010 RETURN
6020 FOR n=1 TO 7
6030 LET d=4+INT (RND*22)
6040 IF ATTR (4,d)=71 THEN GO TO
 6030
6050 PRINT AT 4,d;" "
6055 LET d(n)=d
6060 NEXT n
6070 RETURN
6100 LET a=1: LET a$="          *Mar
tin Kehr zeigt "
6110 PRINT INK 4;AT 10,5;"[illegible]
[illegible]"; INK 5;AT 11,
5;"[illegible]"
6120 PRINT AT 14,6; INK 2;"© 198
4 by Martin Kehr"
6130 PRINT INK 6;AT 17,0;"Steuer
ung efolgt ueber o und p !"; INK
 3;AT 18,3; INK 6;"In Verbindung
 mit C-Shift";AT 19,9;"springen
Sie!"
[illegible] PRINT INK 3;AT 4,11;"[illegible]
[illegible]";AT 6,11;"[illegible]";AT 5,11
;"[illegible]"
6300 PRINT AT 5,12; INVERSE 1;a$
(a TO a+6)
6310 LET a=a+1
6320 IF a+6>LEN a$ THEN LET a=1
6325 PAUSE 7
6330 IF INKEY$="" THEN GO TO 630
0
6340 RETURN
```

Kleinanzeigen

Verkäufe

ATARI 600XL + Floppy preisg. abzugeben. Tel. 06 21/ 56 22 41, am Wochenende

ZX 81, m. gr. Tast., 16 K RAM, 64 K RAM, Drucker ZX, 3 Software-Bücher, Preis VB 570,–, Tel. 0 53 02/18 51

Dragon 32, suche Programme, z. B. Adventures, Simulations-Programme, Action-Games etc., Jun Kurokoshi-Krüger, Kattendahlerstr. 36, 4006 Erkrath

Hallo, Texaner! TI-99/4A Software (Superspiele/Spielegenerator etc.) bei: A. Ertl, Mais 66, 8497 Neukirchen. Liste mit tollen Sachen gegen Freiumschlag! Sofort schreiben.

Plattenarchiv . . . . . . 20,– DM
Diskhilfe . . . . 20,– DM
P. Schulz, Eichendorffallee 6, 3352 Einbeck 1

M. Wolf, Tel. 06 41/4 14 50, tausche ca. 600 Programme

Sinclair ZX Spectrum 48 K bei uns nur DM 448,–, Sinclair QL DM 1798,–, Microdrive u. Interf. je DM 199,– bei Computer-Store, Herzebrocker Str. 46, 4830 Gütersloh, Tel. 0 52 41/1 20 80

Apple II: E. T. (Alleslerner) o. Geldspieler. Tamm, Solmsstr. 43, 1 Berlin 61

Apple II Super-Disketten-K + ility m. fantastischen Möglichkeiten. Info g. Freiumschlag bei: L. Düning, Kurze Brehe 10, 4500 Osnabrück

Verk. ZX 81 (Fol.-Tast. Def.) + 16K + Netzt. + Koaxkabel + Rec.-Kab. + deut. Anltg. + Buch M.80 Prgr., zus. 120,– DM. Chr. Lenk, Langlütjensand 4, 28 HB 66, Tel. 04 21 / 58 73 26

Verk. **Colour-Genie 32K,** mit original Software NW 600,–, 3 Handbüchern u. Datenrecorder für nur 550,–. Tel. 0 47 21 / 4 72 80

Verkaufe Super-Grafik-Abenteuer „Jetset-Freddy". Mit Highscoreliste, 2 Spielstufen und 6 Höhlen (auf Cassette)! 15,– DM an Bernd Schissler, Hölderlinstr. 3, 7117 Bitzfeld

Verk. TI 59 + Drucker PC100C + Anleitungen + Netzteil + 60 Magnetk. + 2 Rollen Papier, zus. für 450,– DM. W. Lenk, Langlütjensand 4, 28 HB 66, Tel. 04 21 / 58 73 26

FORTH-Assemblerlistings für alle wichtigen Mikroprozessoren je 25,– DM; Installation-Manual 15,– DM, Autolinker u. Buchhaltung f. Apple II. Kostenl. Info anf. bei B. Lipgens – Datentechnik, Wilbrechtstr. 65, 8000 München 71

Verkaufe: TI-99/4A + Ext. Basic + Joysticks + Recorder + Kabel + TI-Invader + dt. Handbuch + ca. 70 PGM. + 1 TI-Magazin + Chip-Buch. VB: 700,– DM. Tel.: 02 21 / 60 76 76

Achtung: TI99/4A + Mini Memory, Editor/Assembler/Disassembler. Paket nur 30,– DM in Scheinen an T. Mielke, Falkenkamp 17b, 2000 Norderstedt 1

C-64-Forth. Diese neue Sprache jetzt auch f. **C-64** Cass.-Vers. DM 63,–/76,–. Diskvers. inkl. Handb. (einz. DM 25,–) **D-BASIC,** prof. BASIC erw., DM 33,– D. Luda, Staudinger Str. 65, 8000 München 83

Verk. TI-99/4A + Ex-Basic + Sprach-Synthesizer + Joyst.-Adapter + ca. 100 Prg. + Zubehör. VB. A. Neuburger, Feldstr. 16, 7505 Ettl. 8, Tel. 0 72 43 / 2 91 75.

TI 99/4 A Ext.-Spiel nur für Erwachsene – bitte Alter ang. zum absoluten Preishit von nur DM 10,- im Umschl. an T. Karbach, Remscheider Str. 18, 5650 Solingen 1

Wenn ihr „Mugsy von Melbourne House" wollt, dann kriegt ihr es für 35 DM (vielleicht) noch bei: Jan Weigner, Carl-Schurz-Str. 11, 2800 Bremen 1, Tel. 04 21 / 34 63 87

Wegen Systemwechsel zu verkaufen: **Laser 2001,** Diskettenlaufwerk, Datenrecorder, Printerkabel + Software auf Diskette. VB 1100,–. Tel. 0 61 71 / 7 94 73 ab 18 Uhr

ZX-Spectrum Soft- und Hardware-Neuheiten aus England: Monty Mole, Mugsy, Sabre Wolf usw. + Hardwareartikel. Anfrage lohnt sich. Gratiskatalog bei: Dremasoft, Im Rosenhag 6, 53 Bonn 1

D-64 Forth. *** 49,– DM, Handb. 25,– DM *** Info: D. Luda, Staudinger Straße 65, 8 München 83 ***

★★★ FLOPPY 1541 6 x SCHNELLER ! ★★★ durch Parallelbus für C 64. Interface kpl. geprüft mit Software nur DM 179,–. Info gegen DM 1,50 in Briefmark., B. Akesson, Pf. 802, 4040 Neuss.

Ankäufe

TI-99/4A
Suche **Extended Basic Modul** evtl. Tausch gegen **Editorassembler** sowie Maschinensprachprogramme z. B. **Flugsimulator** und **Ex. Basic Spiel Miner 2049** (USA-Version).
Tel. 0 71 32 / 3 76 08

Suche
VC-20-Programme
Fritz, Wichdorfer Str.14
3501 Emstal 2

Apple II. Suche Software, evtl. auch Tausch. Schickt Eure Liste an Jörn Lubkoll, Boothstr. 23 b, 1000 Berlin 45 *** nur Disk ***

Suche: Gute und billige Software auf Diskette für 64K Atari 600XL. Angebote an: Michael Ebeling, Riepener Straße 7, 3061 Beckedorf

Suche **Atari Diskettenstation,** Aussehen egal, Hauptsache sie funktioniert. Bitte anrufen unter Berlin 0 30 / 4 93 – 19 27

TI 99/4A Minimemory + Logo gesucht. Tel. 08 81 / 6 21 64

C-64. Suche Assembler, Pascal, Forth und andere Programmiersprachen. Tauschmaterial vorhanden. Stefan Schneider, Meerschlader Weg 1, 5275 Bergneustadt

Kontakte

Hallo Spektrumfreaks. Ich möchte mit euch gerne Prog. tauschen. Liste, ob groß oder klein, an: Robert Weidinger, Schönmetzlerstr. 10, 8050 Freising oder Sams.– Diens. 0 81 61 / 43 72

Kontakt mit MZ700-Leuten gesucht. Tel. 05 11 / 42 29 75

***** Atari-Spectrum-User-Club *****
monatl. Info, bundesweit organisiert, riesige Prog.-Bibliothek usw., melden bei:
ASUC, Am Schürhof 17
4905 Spenge
Tel. 0 52 25 / 20 68

Comp.-Club
Bad Hersfeld
sucht Mitglieder!
Kontakt über
Roland Reyer
Am Giegenberg 21
6438 Ludwigsau 1
oder Jugendhaus Hersfeld

Tausch

● ZX-Spectrum ● neue Programme ● Tauschbörse ● Info von S. Sauer ● Heideweg 8, 2720 Rotenburg ●

Tausche/Verkaufe Atari Prg. 16 K. Uwe Schöneberger, Sonnenstraße 18, 6652 Bexbach 2

Suche Software aller Art (Disk.), für Atari 800 XL u. Antic-Hefte bis März 1984, evtl. auch Computer u. Analog. Dietmar Levenhagen, Landwehr 83, 3057 Neustadt 1, Tel. 0 50 32 / 6 11 81

ZX-Spectrum
Software und Interessenaustausch
Monika Baumgartner
4021 Linz
Postfach 142 Austria
Bitte Liste beilegen

Wenn Sie in der Lage sind, auch andere an Ihren Phantasien teilhaben zu lassen, versuchen Sie es doch einfach mal mit selbsterstellten Programmen!

Wir suchen ständig neue, phantastische Abenteuer, spannende und lustige Programme und natürlich auch Anwendersoftware.

Wir brauchen von Ihnen:

1) Ausführliche Spielbeschreibung
2) Datenträger (Kassette oder Diskette)
3) Komplettes Listing (nicht unbedingt erforderlich)

All dies schicken Sie bitte an den Tronic-Verlag, Postfach, 3444 Wehretal 1.

Auftrag für Gelegenheitsanzeigen in Homecomputer

Homecomputer
Landstraße 29
Postfach 41
3444 Wehretal 1

Unter der Rubrik „Kleinanzeigen" veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Preise für „Kleinanzeigen":
Private Gelegenheitsanzeige je Druckzeile 5,— DM inkl. MwSt.
Chiffregebühr je Anzeige 10,— DM.
Gewerbliche Gelegenheitsanzeige je Druckzeile 11,— DM inkl. MwSt. (dürfen nicht unter Chiffre erscheinen).

Name und Adresse

Abo-Nr.

Unterschrift | Ich zahle sofort nach Rechnungserhalt | Datum

Ich wünsche folgenden Text zu veröffentlichen:

Zutreffendes ankreuzen

☐ Ankäufe ☐ Kontakte ☐ Geschäftsempfehlungen ☐ gewerblich
☐ Verkäufe ☐ Tausch ☐ Verschiedenes ☐ privat

Und ab geht die Post . . .

HOMECOMPUTER neuer Kassettenservice '85

Laser 2001

Laser 2001 K = 11,00
Clipper-Flipper

ZX 81

ZX 81 K = 10,00
Grand Prix
Irrgarten

C 64

C 64 K = 17,50
Reactor D = 23,50
Concentration
Datenbank

Apple

Apple K = 14,50
D = 19,50
Donovan
Basic-Konverter
Maskengenerator



TI 99

TI 99 K = 14,50
D = 19,50
Liftbär
Luftkampf

Atari

Atari K = 14,50
Maniac D = 19,50
Fünf gewinnt

ZX-SP

ZX-Spektrum K = 13,50
Plong

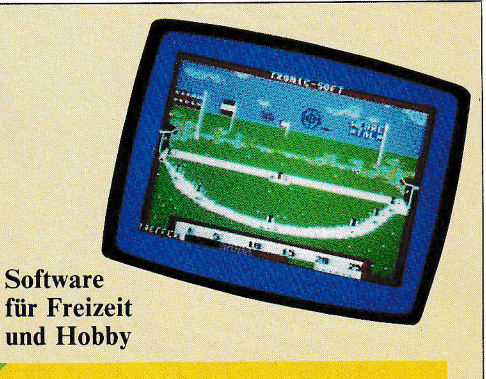
VC 20

VC 20 K = 11,50
D = 19,50
Zyklo

Apple

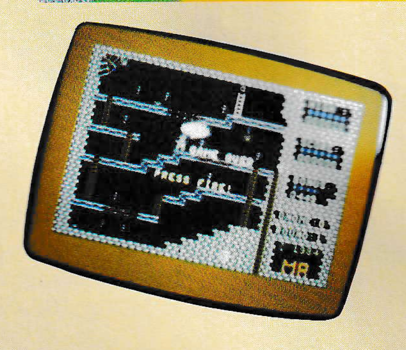
Apple K = 14,50
D = 19,50
Labyrinth
Data-Generator

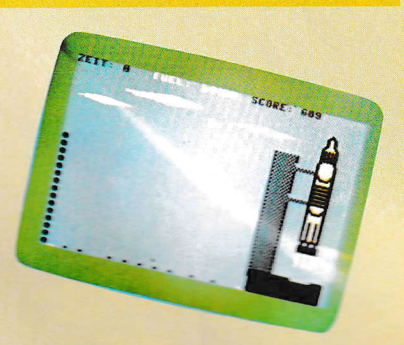
Kassettenservice



Software
für Freizeit
und Hobby







HEFT 9

Colour Genie K = 10,--DM
Fuchs und Hund

C-64 K = 16,--DM
D = 23,50DM
Projekt
Datenbank

Atari K = 14,--DM
D = 19,50DM
The Big Quest
Fünf gewinnt

Apple D = 19,50DM
Diamonds
Hilfsprogramm

TI 99 K = 14,50DM
*D = 19,50DM
Transfer
Silverspar

Laser 2001 K = 12,--DM
Crazy Cake

ZX-81 K = 10,--DM
Reversi

ZX-Spectrum K = 14,50DM
Jump about

VC-20 K = 14,--DM
D = 19,50DM
Garten
Schloß Gruselstein

Dragon K = 10,--DM
Anwenderprogramm

HEFT 10

Ti-99 K = 14,50
D = 19,50
Mother-Duck
Screen-Designer

VC-64 K = 16,50
D = 23,50
Spiders
The Basic

Atari K = 11,--
D = 18,50
Splitt

ZX-Spectrum K = 14,50
Pac-Man
Oil-Panic

VC-20 K = 14,--
D = 19,50
Fressmann
Outlaw

Apple D = 19,50
Tic - Tac - Toe
Jumper

ZX-81 K = 10,--
Panik Labyrinth

Dragon 32/64 K = 14,--
Dragon paint

Laser 2001 K = 8,50
Cave-Man

HEFT 11/12

VC-64 K = 17,50
D = 23,50
High Noon
Skeet
Grafik-Designer

TI-99 K = 14,50
*D = 19,50
Cave-Man
Alkoholverbot

Atari K = 14,50
D = 19,50
Ski
Mutation

ZX-Spectrum K = 16,--
Frogger

ZX-81 K = 10,--
Expedition

VC-20 K = 14,--
D = 19,50
Prost
Buffalo Bill

Apple K = 14,50
D = 19,50
Donovan
Basic-Konverter



Das neue Abenteuerspiel von Atari
"Das Geheimnis von Atlantis"

Die sagenumwobene Stadt wartet darauf entdeckt zu werden.

Ein Forscher findet in einer alten Bibliothek Unterlagen über die sagenumwobene Stadt ATLANTIS. Ein unwiderstehlicher Drang läßt ihn keine Minute mehr zur Ruhe kommen, um das damit verbundene Geheimnis zu lösen. In Erwartung auf Ruhm und Reichtum verkauft er alles, was er besitzt, um die Expedition auszurüsten und zu finanzieren.
Der Spieler jetzt in der Rolle des Forschers, befindet sich in seinem Boot auf dem Meer, irgendwo dort, wo der Legende nach der Kontinent untergegangen sein soll.
Es gilt, sich auf alle Gefahren und Hindernisse vorzubereiten und sich entsprechend auszurüsten. Ein reichliches Sortiment an Werkzeugen und Hilfsmitteln findet der Spieler in der Kabine. Jetzt heißt es, die richtige Wahl zu treffen z. B. zwischen Anzug, Lampe, Atemgerät und anderen Gegenständen.
Aber Vorsicht, man kann nicht alles tragen.
Nun beginnt das Unterwasserabenteuer.
Zahlreiche Aufgaben liegen vor ihm, wobei nicht nur an die Überwindung von Hindernissen gedacht werden sollte. So manche Sache auf dem Meeresgrund ist später eventuell verwertbar.
Eine liebliche Nixe, die seinem Weg kreuzt, bleibt nicht seine einzige Begegnung. Unerfreulich und gefährlich sind die Kontakte mit Quallen und Schlingpflanzen.

Nach einem langen, mühsamen Weg ist dann endlich ATLANTIS erreicht, wo das eigentliche Abenteuer beginnt. Dort heißt es, die Suche nach dem Schatz aufzunehmen, der in den Mauern von ATLANTIS verborgen liegt.
Dies jedoch zu erzählen, hieße die Spannung vorwegzunehmen ....

Brigitte und Günther Möhle, die Programmierer haben sich mit ATLANTIS ein wirklich fesselndes Spiel ausgedacht. Neben den vielen Spielvarationen bestechen bei ATLANTIS besonders Gestaltung und Farben der Grafiken.
"ATLANTIS" ist für ATARI Computer Systeme und Commodore C-64 ab Mitte November 1984 erhältlich und erfordert mindestens 48K RAM.
"ATLANTIS" ist zu einem Preis von DM 69,- (unverbindliche Preisempfehlung) zu haben.